Burroughs

B 1700 On-Line Data Entry System (ODESY)

USER INFORMATION MANUAL

Copyright © 1977, Burroughs Corporation, Detroit, Michigan 48232

Printed in U.S.A.

July 1977

2011771

LIST OF EFFECTIVE PAGES

NOTE: Insert latest changed page;
dispose of superseded pages.

TOTAL NUMBER OF PAGES IN THIS MANUAL IS 328 CONSISTING OF THE FOLLOWING:

Page No.	Issue
Cover	Original
Inside cover	Blank
Title	Original
A	Original
i thru ix	Original
x	Blank
1-1 thru 1-14	Original
2-1 thru 2-20	Original
3-1 thru 3-15	Original
3-16	Blank
4-1 thru 4-49	Original
4-50	Blank
5-1 thru 5-57	Original
5-58	Blank
6-1 thru 6-9	Original
6-10	Blank
7-1 thru 7-3	Original
7-4	Blank
8-1, 8-2	Original
9-1, 92	Original
10-1 thru 10-37	Original
10-38	Blank
A-1	Original
A-2	Blank
B-1	Original
B-2	Blank
C-1 thru C-30	Original
D-1 thru D-7	Original
D-8	Blank
E-1 thru E-7	Original
E-8	Blank
F-1 thru F-25	Original
F-26	Blank
G-1 thru G-3	Original
G-4	Blank
H-1 thru H-11	Original
H-12	Blank
I thru IX	Original
X	Blank

Burroughs believes that the information described in this manual is accurate and reliable, and much care has been taken in its preparation. However, no responsibility, financial or otherwise, is accepted for any consequences arising out of the use of this material. The information contained herein is subject to change. Revisions may be issued to advise of such changes and/or additions.

Table of Contents

Section	Title	Page
	INTRODUCTION	vii
1.	SYSTEM CAPABILITIES AND CONCEPTS	1-1
	1.1 GENERAL	1-1
	1.2 SCREEN DISPLAY, INPUT AUDIT, OUTPUT AND JOURNAL FACILITIES	1-1
	1.3 SYSTEM SOFTWARE	1-4
	1.4 HARDWARE	1-6
	1.5 DATA CONTROL	1-7
	1.6 RECOVERY	1-8
	1.7 SECURITY	1-9
	1.8 EMERGENCY PROCEDURES	1-10
	1.9 INQUIRY FACILITIES	1-12
	1.10 RUNNING OTHER ON-LINE PACKAGES	1-13
2.	INSTALLATION PROCEDURES	2-1
	2.1 IMPLEMENTATION PLAN	2-1
	2.2 SELECTING SYSTEM FEATURES	2-3
	2.3 GENERATION OF THE NETWORK CONTROLLER	2-9
	2.4 FIELD ENGINEER SETTINGS FOR TD820, TD830 AND TD730 TERMINALS	2-11
	2.5 STANDARD SYSTEM CONFIGURATION	2-12
	2.6 FILE SIZES	2-14

Table of Contents

Section	Title	Page
2. (Cont'd)		
	2.7 RUNNING ODESY INITIALLY	2-16
	2.8 RECONFIGURATION OF THE ODESY SYSTEM	2-18
3.	THE FILEMAINT PROGRAM	3-1
	3.1 GENERAL	3-1
	3.2 BLOCK DIAGRAM	3-3
	3.3 INITIALIZATION OF THE FILES	3-4
	3.4 CONFIGURATION PARAMETERS	3-5
	3.5 STATION PARAMETERS	3-9
	3.6 OPERATOR PARAMETERS	3-10
	3.7 ERROR TEXTS	3-11
	3.8 COMMANDS	3-12
	3.9 CHECK DIGIT VERIFICATION	3-13
	3.10 PRINT FACILITIES	3-14
	3.11 SQUASH FACILITIES	3-15
4.	FORMAT MAINTENANCE	4-1
	4.1 GENERAL	4-1
	4.2 BLOCK DIAGRAM	4-3
	4.3 FORMAT MAINTENANCE COMMANDS	4-4
	4.4 ENTRY OF FORMAT DATA	4-14

Table of Contents

Section	Title	Page
4. (Cont'd)		
	4.5 GROUP HEADER DETAILS	4-16
	4.6 FORMAT HEADER DETAILS	4-20
	4.7 FIELD DETAILS	4-23
	4.8 CONSTANT DETAILS	4-36
	4.9 WORKED EXAMPLE 1	4-39
	4.10 WORKED EXAMPLE 2	4-44
5.	THE EDIT PROGRAM	5-1
	5.1 GENERAL	5-1
	5.2 BLOCK DIAGRAM	5-4
	5.3 DATA ENTRY	5-5
	5.4 OPENING A BATCH	5-7
	5.5 DATA VERIFICATION	5-9
	5.6 RELEASING A BATCH	5-12
	5.7 BATCH CORRECTION AFTER RELEASE	5-15
	5.8 BATCH TOTALS	5-16
	5.9 USE OF INQUIRY MODE	5-17
	5.10 EXPLANATION OF INDIVIDUAL COMMANDS	5-18
	5.11 BASIC DATA ENTRY - WORKED EXAMPLE	5-51

Table of Contents

Section	Title	Page
6.	THE EXTRACTION PROGRAM	6-1
	6.1 GENERAL	6-1
	6.2 BLOCK DIAGRAM	6-2
	6.3 EXTRACTION AND RELEASE	6-3
	6.4 DELETION	6-5
	6.5 STATISTICS	6-6
	6.6 RECOVERY	6-7
	6.7 ON-LINE AND STAND-ALONE VERSIONS	6-8
	6.8 JOURNAL-ONLY FACILITY	6-9
7.	THE STATISTICS ANALYSIS PROGRAM	7-1
	7.1 GENERAL	7-1
	7.2 BLOCK DIAGRAM	7-3
8.	THE BATCH STATUS PRINT PROGRAM	8-1
	8.1 GENERAL	8-1
	8.2 BLOCK DIAGRAM	8-2
9.	AUXILIARY PROGRAMS	9-1
	9.1 GENERAL	9-1
	9.2 TANK FILE ANALYSIS PROGRAM (DUMP)	9-1
	9.3 DUMPFILE CONVERSION PROGRAM (CONVERT)	9-1

Table of Contents

Section	Title	Page
10.	INQUIRY FACILITIES	10-1
	10.1 GENERAL	10-1
	10.2 USER PROGRAM INTERFACE	10-4
	10.3 INQUIRY FORMATS	10-11
	10.4 INQUIRY MODE	10-14
	10.5 USE OF DATA ENTRY COMMANDS WITH INQUIRY FORMATS	10-15
	10.6 WORKED EXAMPLE 1	10-17
	10.7 WORKED EXAMPLE 2	10-24
APPENDIX A	DATA ENTRY AND CONTROL COMMANDS	A-1
APPENDIX B	FORMAT MAINTENANCE COMMANDS	B-1
APPENDIX C	INPUT FORMS	C-1
APPENDIX D	OUTPUT REPORTS	D-1
APPENDIX E	CHECK DIGIT SCHEMES	E-1
APPENDIX F	ERROR AND INFORMATION MESSAGES	F-1
APPENDIX G	INTERNAL AND EXTERNAL FILE NAMES	G-1
APPENDIX H	GLOSSARY OF TERMS	H-1

INTRODUCTION

The B1700 On-line Data Entry SYstem (ODESY) is designed to facilitate the entry of data into a computer by using multiple on-line visual display units for both entering the data and for verifying it after entry. It will thus be possible to ensure that all data (which is usually entered in batches) is correct before it is processed by a user application system. To this end numerous input audit checks and display options are provided by the system. ODESY also allows reformatting options to be used, so that batches of output data may be correctly formatted for use with a user application program, and incorporates journal facilities to provide hard copy of system details and data entered.

By eliminating the need to use punched cards, which have long been the traditional medium for data entry, ODESY could also eliminate the need for a separate data preparation department by allowing the VDU's to be placed in the departments from which the data originates. This means less time needed for data preparation, less expense, and less risk of errors both because there are fewer stages in the data preparation process and because those most familiar with the meaning of the data will be those who are actually entering it.

The advantages of using the ODESY system can be summarized as follows :-

1. The data is entered by staff familiar with its meaning.

2. The wide range of checks ensures correctness of data prior to extraction.

3. The system is flexible; the number and type of terminals can be varied.

4. The system can be run on existing mainframe and terminal equipment.

5. The system can be used as a 'front end' input control system when designing future application systems.

ODESY is intended for users of B1700 machines who require an on-line data entry system and either already have the necessary terminals or are intending to acquire them.

The manual may be used by management, systems analysts, programmers and operators and is designed as a general guide to ODESY, its structure and its features. It contains the information necessary to set up the system and to keep it running smoothly once it has been set up.

There are other manuals relevant to ODESY which cater for specific staff groups.
These are :

ODESY Console and Terminal operator's Manual 2011763 → THIS HAS BEEN REPLACED BY # 1131851

TD 700/800 Equipment Reference Manual 1068699

B1700 System Software Operational Guide 1068731

B1700 Systems Network Definition Language (NDL) ref. man. 1073715

TD820 Equipment Reference Manual 1085685

TD730/TD830 System Reference Manual 1093788

1. SYSTEM CAPABILITIES AND CONCEPTS

1.1 GENERAL

To enter data into ODESY the user must first set up a number of formats. These are displayed on the screen and data is entered into the individual fields in the format. Section 4 gives a detailed description of setting up and maintaining formats and Section 5 deals with the entry of data into the formats.

1.2 SCREEN DISPLAY, INPUT AUDIT, OUTPUT AND JOURNAL FACILITIES

The system includes a comprehensive range of options giving control over exactly how the formats appear, either when displayed or when output for extraction, how many and what sort of checks are carried out on the data, and also whether or not it is required to print output records in reports, either as a batch or individually as selected records.

The facilities offered can be divided into 4 different types of option. These are as follows:

1. Screen display options
2. Input audit check options
3. Output options
4. Journal options

The screen display options govern how the format appears when actually displayed on the terminal screen. They offer the following facilities:

- constants or default values may be inserted in any field in a record.
- fields may be duplicated from record to record.
- optional new line after a field for screen formatting.
- optional 10 character field identification may precede each field.
- following formats may be automatically displayed on the successful entry of data.
- data may be displayed or suppressed during verification.
- error count and field(s)-in-error indicator are provided on redisplay of incorrect data.

The input audit check options provide a means of ensuring that data in each format is as correct as possible before it is scheduled for release. These are:

- type either alphanumeric or (signed) numeric.
- mandatory left or right justification.
- check digit with variable moduli and weighting.
- mandatory or optional field.
- range checks on numeric fields.
- value checks against a constant list.
- batch control totals.
- crossfoot register.

The output options govern how the data appears in the output records. They provide the facility to convert data types and also to reorganise records. They can be summarised as follows:

- ability to move fields to any position in the output record.
- data conversion from type character to packed.
- automatic sign insertion (leading, trailing or overpunch) for numeric fields.
- justification of fields to the left or right.
- zero or blank fill.
- multiple output records from one input record.
- a single output record from multiple input records.
- choice of output media
    a. disk
    b. magnetic tape
- choice of physical file attributes
    a. file name
    b. record length
    c. blocking factor
    d. records per area
    e. number of areas
    f. label type (Burroughs, ANSI)

The journal options enable the user to specify a report (journal) listing the output records. Selected records and selected fields within each record

may be chosen for printing. Fields are identified by their descriptions if requested.

More detailed descriptions of all the options available, and of how they may be used, appear in the chapter on format maintenance.

1.3 SYSTEM SOFTWARE

There are three files in the system with which the user will need to be familiar. These are the System file, which contains details of operators, stations etc, the Formats file, which contains the user-created formats and the Tank file which is a work file used during data entry. Each of these files is maintained by a different ODESY program. The programs in the system are as follows:

The FILEMAINT program

This program is used to initialise and create the Tank, Formats and System files. In addition it maintains the various records of the System file which may be altered by the user.

The FORMAINT program

This program is an on-line interactive program by means of which the user may create input, output and journal formats to his own requirements. The

program uses prompts and error displays at the VDU to ensure that formats are created correctly. Note that format maintenance may only be performed from ONE terminal and may not be run concurrently with data entry. The Formats file is maintained by this program.

The EDIT program

This is also an on-line program which performs the actual collection of data from the various data entry stations. It checks and verifies the data, schedules batches for release and also performs several types of auxiliary functions such as inter-station message switching, debugging traces, dumps and program termination. The Tank file is maintained by this program.

The EXTRACT program

This program is used to convert data from the format in which it is held in the Tank file to a format which is suitable for use in a user application program. There is a wide range of output format options available to the user.

The STATS program

During the running of the EDIT and EXTRACT programs, statistics are accumulated by format and by operator. This program clears these statistics and prints them out in the form of a report.

A network controller (handler) is also required. This may be quickly and easily generated by using Burroughs Network Definition Language (NDL).

The PRINT program

This program prints the current status of all batches in the Tank file.

1.4 HARDWARE

The system is designed to run on a Burroughs B1700 computer and requires at least 64kb of memory, some of which will be available for multiprogramming. In addition to the B1700 central processing unit, the minimum hardware configuration necessary to run the system is:

1. One single line control.
2. One line adaptor (preferably 9600 BPS).
3. At least one TD700 or TD800 style terminal.
4. One line printer and control.
5. 4.6 MB Disk.

The amount of memory required for the system will depend on both the hardware configuration being used and also on the software options. It is possible to optimise memory requirements for the EDIT program by specifying what commands are required, how many formats will be used etc. This is discussed at greater length in Section 2.

1.5 DATA CONTROL

As has been previously mentioned, ODESY allows the user to dispense with a central data preparation department and enter the data in the department where it originates. Firstly, this considerably reduces the number of errors since the staff entering the data will be familiar with its meaning. Secondly the fact that there is less transcription of data will reduce the amount of work involved in the actual physical data control process.

ODESY has numerous features which ensure that the data entered is quickly and easily identifiable and also as correct as possible before extraction takes place.

Data is entered into the system, verified and released in batches. If verification is to be used, it is recommended that a batch is entered by one operator and verified by another so as to reduce the possiblity of errors being repeated. However the user may find that the error checking facilities provided during data entry make verification unnecessary. How large the batches are will be decided by the user. It will probably be found that it is best to start with small batches and increase the batch size as one becomes more familiar with the operation of the system.

Whilst it is not essential to have a batch header, it is preferable to have documents grouped together in batches with the first record as the batch

header. Batch numbers may or may not be allocated by the user. Where the computer has allocated the batch numbers these should be noted on the input forms by the operator.

1.6 RECOVERY

The ability to recover system operation after a machine fault has occured, and to ensure the integrity of the files produced is one of the most important aspects of any real-time system. ODESY contains, as part of the EDIT, a recovery system which automatically recognises that the system has terminated abnormally and restores it to the exact state that it was in prior to termination.

The maximum amount of data in each batch that may be lost in such a case is the last incoming message currently being processed by the system. This never amounts to more than 1 record per work station.

The terminal operator will merely receive the standard greetings message to inform him or her that recovery has taken place and he or she is then free to continue entering data.

The central machine operator just restarts the EDIT and the recovery system will perform all the necessary operations to restore the files. No dumping of files is necessary.

1.7 SECURITY

ODESY contains a number of security checks which are designed to prevent unauthorised access to the system and limit access to certain formats to certain staff only.

Each operator or supervisor is assigned a unique operator number and a password. These are used for logging into the system and if the number and password have not been previously defined in the System file, access will be denied. This is also the case when the correct number but the incorrect password is keyed in and vice versa.

An operator may be defined as either an operator or a supervisor. There is also a security level assigned to each operator or supervisor which is compared by the system to the format security level. Only if the operator/supervisor security level is equal to or greater than the format security level may this operator access this format. The user should have no difficulty in arranging the hierarchy of operators/supervisors and security levels to fit his own requirements.

There is also the facility to restrict the use of certain commands such as RELEASE or ACCEPT to supervisors. This provides the supervisor with a greater degree of control over the data entry procedure.

1.8 EMERGENCY PROCEDURES

There are two types of systems failure for which provision should be made. These are:

1. Hardware failure
2. Software failure

Hardware failure can be (a) a fault in the central processing unit or its peripheral devices, or (b) a fault in the line to the terminal or (c) a fault in the terminal itself. The NDL will detect the majority of line and terminal faults but the user might like to have Burroughs DC/AUDIT program available to test out the hardware for faults.

If a terminal cannot be used because of a terminal fault or a line fault, an engineer will be required to rectify the fault and the operator may resume when this has been done. Should the operator wish to continue elsewhere, he/she may log out at the failed terminal by entering /BYE <operator number> at the new terminal. The operator may then log in at the new terminal and resume the entry of data into the same batch, if desired. No data will have been lost.

If the central processor fails the recovery system will again ensure that when the fault is corrected and the system restarted, only a minimal amount of data will have been lost. A more serious case for which it is necessary to have backup tapes or backup disks may arise when the disk(s) on which

the data entry system files reside become corrupted. It may be possible to delete the records in error and reinsert them again if only slight corruption has occurred. However, in the event of disk failure where the files have become badly corrupted or unreadable, it is necessary to use existing backup procedures to copy the files from tape or disk. How much data is lost will depend on the frequency of the user's backup procedures. It is recommended that disk files are dumped to tape at least once a day if the system processes a large volume of data.

1.9 INQUIRY FACILITIES

ODESY provides a number of inquiry facilities. These are described in detail in section 10.

If the inquiry facilities are required, then it is necessary for the user to write a program, which will access a data base or master file in response to requests from the Edit. The user program has two basic functions: it supplies data to the Edit, and it checks (or stores) data from the Edit. Because the data base is accessed by the user program, there are no restrictions on the type of data base management system that is employed, and the user has complete control over what is done.

These inquiry facilities are intended to supplement the batch processing facilities available in ODESY. The entry of data may be speeded up if some of the data is supplied from the data base rather than entered by the operator, and errors should be reduced if the data entered is validated against the data base. However, it is also possible to use the facilities for on-line updating of a data base or for simple information retrieval.

WARNING: IF THE ONLY REQUIREMENTS OF THE USER ARE FOR ON-LINE UPDATING OR SIMPLE INQUIRY, AND HE HAS NO NEED FOR BATCH PROCESSING, THEN THE USE OF ODESY IS NOT RECOMMENDED.

1.10 RUNNING OTHER ON-LINE PACKAGES

The user may wish to run other on-line packages at the same time as ODESY. This can be done in one of two ways, depending on how the network is to operate.

If each station in the network is dedicated to one application, then it is only necessary to set up the NDL specifications correctly. For instance, in a network which is to run ODESY and CANDE simultaneously, the ODESY stations should appear in the remote file STATIONS, and the CANDE stations should appear in the remote file CANDEQUEUE. Problems will only occur if the two remote files have stations in common.

If each station in the network is to be able to select its own application, the network needs to be configured in a different way. The Edit is an MCS (Message Control System), and packages such as CANDE and GEMCOS also use an MCS. In order to transfer stations between MCS-type programs, it will be necessary to use the dynamic attach/detach procedures available to an MCS.

The simplest way to control the network is to use a Supervisory MCS. This Supervisory MCS will ZIP-execute the other programs, dynamically attach stations to programs, and generally control the network. It will need a station table, a program table and a remote file table. It should open a remote file SUPERQUEUE containing all the stations of the network, and the other remote files (STATIONS, CANDEQUEUE, etc) should be dummy remote files. The Supervisory MCS must ZIP-execute ODESY/EDIT, because it needs to know the remote file number associated with the file STATIONS, and the ZIP-execute results in the Supervisory MCS being asked to approve the open of the remote file. The ZIP-execute can be done in response to a

/cont'd

1.10 message such as *EX ODESY/EDIT from a station. The Supervisory MCS can then attach stations to ODESY. Each station wishing to be attached would send a message such as *ATTACH ODESY/EDIT. When performing the attach, the Supervisory MCS should be non-participating, and a signal character (for instance *) should be declared, so that event messages such as *DETACH go to the secondary remote file SUPERQUEUE rather than the primary remote file STATIONS.

2. INSTALLATION PROCEDURES

2.1 IMPLEMENTATION PLAN

The time required to implement the On-line Data Entry System will vary from installation to installation but the tasks which must be completed are the same in almost every case. The following list covers those tasks:

1. The available manuals should be read by all necessary people, i.e. data processing manager, chief analyst and analyst in charge of the implementation of ODESY.

2. Arrangements should be made for all necessary hardware to be installed (single or multi line control, VDUs, TDI adapter, extra memory).

3. Using the following sections of this chapter and the forms in Appendix C, specify the system configuration you require i.e. specify the parameters for FORMAINT, EDIT and EXTRACT, define the stations, operators and check digit schemes you require, and modify any texts and commands as necessary.

4. Reconfigure the standard ODESY system using the EDIT, EXTRACT and FILEMAINT programs.

5. Design the formats you will require using the forms in Appendix C, and if the inquiry facilities are required, design the user program (or programs).

6. Set up and test formats using the FORMAINT program and the EDIT. Make any necessary changes after running a few test batches through. It will probably be best to try a few first to learn how to use the format and field options.

7. Train the console and data entry operators in the use of the system. It is important that after a training class, adequate "hands on" experience is provided. Data entry operators in particular should have time to work with ODESY and become proficient at it before parallel running.

8. For some period of time ODESY should produce batches to be processed in parallel with the existing method of data preparation. This will provide the means to compare the accuracy and throughput of the system and an additional period of training for the operators.

9. After a period of parallel running, during which everyone concerned with data preparation should become proficient in using ODESY, the system will then go live, providing the primary method of data entry.

10. As new applications are developed they can be designed to use ODESY as their input control system, freeing data preparation from the 80 or 96 column card format and eliminating the need to develop specialized "front ends" for each new application.

2.2 SELECTING SYSTEM FEATURES

Three of the ODESY programs, EDIT, FORMAINT and EXTRACT, use dynamic memory management. This means that the sizes of certain tables and file buffers used by the programs is determined at execution time. In practice it results in the programs using the minimum memory required for the system configuration at that time, and allows the user to change the system configuration without recompiling any of the programs.

Section 3.4 contains details of how to complete the forms for system configuration details and Appendix C1 contains examples of the forms. The system configuration options which the user may alter are as follows:

Record Size for the Tank File

This option controls the length of a record in the Tank file, and also indirectly controls the largest number of characters which can be accepted from one format during data entry, because the Record Size option must be at least 20 bytes greater than the Input Data Size. For instance, if the option is set to A the record size would be 180 bytes, and the Input Data Size could not be more than 160 characters, which means the Edit could only store information corresponding to two 80-column card images per format.

A = Default value

S: system - SYS CONF

The option also has an effect on the maximum number of stations allowed. A recovery record is written to the Tank file containing information about each station, and so if the record size is larger, more stations may be handled. For instance, if 25 stations were required, the option would have to be set to B, even if 160 characters of input data per format was sufficient.

Three record sizes are available, and are shown in the table below:

Option Setting	Actual Tank Record Size	Maximum Value of Input Data Size	Maximum Number of Stations
A	180 bytes	160 characters	17
B	540 bytes	520 characters	56
C	1080 bytes	1060 characters	115

Input Data Size

This option controls the largest number of characters which can be accepted from one format during data entry. Its value determines the size of one of the tables in the Edit, so that the memory requirements of the Edit may be reduced by minimising the value given to this option. Its maximum value is dependent upon the record size of the Tank file, as can be seen in the table above.

Number of formats, number of fields, number of characters of constant data (for the Edit)

The values given to these three options determine the size of certain tables in the Edit, and hence affect the memory requirements of the Edit. If a low value is given to these three options, the Edit will have to spend more time overlaying formats in memory. On the other hand, a high value will result in the Edit requiring more memory to run. The user should first determine which formats will be in regular use at any one time. This should give a value for the number of formats option. The user should then inspect the Format Maintenance listing in order to determine how many fields and characters of constant data those formats contain, and so derive values for the other two options. The user should be able to determine the optimal size of the tables with a little experience.

Number of fields, number of characters of constant data and scratchpad constant data (for FORMAINT)

The values given to these three options determine the size of the tables in FORMAINT, and hence the memory requirements of that program. However they also determine the size of the formats to be maintained. The number of fields option determines the maximum number of fields per format. The number of characters of constant data option determines the maximum number of characters of constant data per format. The scratchpad constants option determines the maximum number of characters of constant data per field. The

optimal size of the tables will become apparent as formats are being designed.

Station buffer size (for FORMAINT)

It is assumed that only one terminal will be used for format maintenance, since FORMAINT will only allow a single user to change formats at one time. If the terminal has a small screen, then the size of the station buffer need only be as large as that screen, so that the memory requirements of FORMAINT may be reduced.

Number of formats, number of fields, output record size (for EXTRACT)

The values given to these three options determine the size of the tables in EXTRACT, and hence the memory requirements of the program. The formats held in the tables at any one time will all belong to one group, so the sizes of the format and field tables will probably be smaller than for the Edit. The user should determine which formats in each group will be in regular use, and use the largest group to give values to the first two options. The third option controls the size of the output buffer. For instance, if the user is only producing 80-column card images, then the option may be set to 80, and so reduce memory requirements.

Command Inclusion

Certain commands, (and functions such as batch totalling, crossfooting and CDV checks), may be included or excluded as desired. This only affects the Edit. Excluding a command does not reduce the size of the Edit. However, the user may have good reasons for preventing certain commands from being performed. For example, by excluding the CONTROL command, ODESY may only be shut down from the B1700 console.

Command Restriction

Instead of excluding a command, it may be desirable to restrict the use of that command to supervisors only. Once more, this only affects the Edit.

The RELEASE command has three restrict options. It is possible to restrict the command entirely to supervisors, or to restrict only RELEASE DELETE commands, or to restrict only extended RELEASE commands. If the second case is selected, only supervisors are able to delete batches, but any operator can schedule a batch for extraction.

Duplicate Anything

This option controls the effect of the Duplicate In field option during data entry. If this option is not set, then data will only be duplicated on the screen when the previous record was entered using the same format. This is

expected to be the normal requirement. If the option is set then the data on the screen will always be duplicated from record to record, regardless of the format of the previous record.

Number of User programs (for EDIT)

The value given to this option determines the size of one of the tables in the EDIT. The table should be large enough to contain entries for all user programs which are to be run at the same time. If there is never more than one user program executing, then the option may be given a value of one. If the inquiry facilities are not required, the option may be given a value of zero.

2.3 GENERATION OF THE NETWORK CONTROLLER

Before running ODESY it is necessary to generate a network controller using NDL. For the standard system configuration the name of the network controller should be TEXT/HANDLER.

The NDL specifications will vary from site to site. An example is shown here with two TD800's and one TD700 on a single line. The file STATIONS should contain all stations which are to use ODESY.

The terminal buffer size (see BUFFERSIZE statement in TERMINAL section) should be large enough for the largest screen format + 20 bytes for control characters. For a TD802, with a maximum screen size of 24 lines of 80 characters, it is unnecessary to set the terminal buffer size to greater than 1940, and if no format uses more than half of the screen, it could be set to 1000. In the example, it is given a value of 2010, which is more than sufficient.

Those users who are familiar with the program TEXT/EDITOR will know that it too uses a network controller called TEXT/HANDLER. That program opens a remote file called TDFILE1, which is shown in the example. If you are already using TEXT/EDITOR, it will only be necessary to make minor changes to your existing NDL specifications.

For further information refer to the B1700 NDL Reference Manual

```
00001   DECLARATION:%
00002      NIF = "TEXT"/"NIF".%
00003      MAX TALLY [1].%
00004   £NC LIST
00005   £LIBRARY POLLTCTD
00006   £LIBRARY SELECTCTD
00007   £LIBRARY AUTOPOLCTL
00008   £LIST
00009   TERMINAL TD800:%
00010        REQUEST = POLLTCTD : RECEIVE, SELECTCTD : TRANSMIT.%
00011        BUFFERSIZE = 2010.%
00012        ADDRESS = 2.%
00013        TRANSMISSION = 0.%
00014   TERMINAL TD700:%
00015        REQUEST = POLLTCTD : RECEIVE, SELECTCTD : TRANSMIT.%
00016        BUFFERSIZE = 325.%
00017        ADDRESS = 2.%
00018        TRANSMISSION = 0.%
00019   STATION DEFAULT TD8:%
00020        MYUSE = INPUT, OUTPUT.%
00021        RETRY = 5.%
00022        CONTROLLER = FALSE.%
00023        TERMINAL = TD800.%
00024   STATION DEFAULT TD7:%
00025        MYUSE = INPUT, OUTPUT.%
00026        RETRY = 5.%
00027        CONTROLLER = FALSE.%
00028        TERMINAL = TD700.%
00029   STATION TD801:%
00030        DEFAULT = TD8.%
00031        ADDRESS = "1B".%
00032   STATION TD802:%
00033        DEFAULT = TD8.%
00034        ADDRESS = "2B".%
00035   STATION TD701:%
00036        DEFAULT = TD7.%
00037        ADDRESS = "7A".%
00038   LINE SLCLINE:%
00039        ADDRESS = 7:14:0.%
00040        AUTOPOLL = 15.%
00041        CONTROL = AUTOPOLCTL .%
00042        STATION = TD801, TD802, TD701.%
00043   FILE TDFILE1:%
00044        FAMILY = TD801.%
00045   FILE TDFILE2:%
00046        FAMILY = TD802.%
00047   FILE TDFILE3:%
00048        FAMILY = TD701.%
00049   FILE STATIONS:%
00050        FAMILY = TD801, TD802, TD701.%
00051   FINI
```

Figure 2.1 A typical network controller

2.4 FIELD ENGINEER SETTINGS FOR TD820, TD830 AND TD730 TERMINALS

For TD820, TD830 and TD730 terminals the following field engineer settings are required:

1. Enable Programmatic Mode Control
2. Disable Field Overflow Inhibit
3. Disable Tab Field Identifier

It is also recommended that the following be done:

4. Disable CR Symbol display
5. Disable ETX Symbol display

2.5 STANDARD SYSTEM CONFIGURATION

The ODESY system is supplied with a standard set of files which will enable the user to run any of the ODESY programs as soon as a network controller has been generated. The file ODESYDATA/SYSTEM contains a standard system configuration which should be adequate for preliminary testing or demonstrations. The file ODESYDATA/FORMATS contains the standard system formats referred to in Section 3 and Appendix C. The file ODESYDATA/TANK contains one active batch, batch 999, which holds the data to enable the system configuration to be quickly modified.

The standard system configuration contains one station, eleven operators, three CDV schemes and all the standard error texts and commands. The eleven operators are numbered 1 to 5 (with passwords SUP1 through SUP5), and 10 to 15 (with passwords OP10 through OP15). The three CDV schemes are numbered 1 to 3, and are Burroughs standard Modulus 10 (same as IBM standard Modulus 10), IBM Modulus 11 and Burroughs Modulus 11 respectively.
The station is specified as logical station number 1 (lsn 1). In the standard system configuration it is defined as a TD802 with braces as forms delimiters. For testing purposes, this definition is compatible with TD701, TD702 and TD801 terminals provided they use braces as forms delimiters. If you are using brackets as forms delimiters, then it is necessary to run FILEMAINT to change the definition:

```
EX ODESY/FILEMAINT; FILE INPUT NAME ODESYSETUP/BRACKETS
```

If you have TD820, TD830 or TD730 terminals, then it is also necessary to run FILEMAINT, but a different input file is required (also refer to section 2.4).

```
EX ODESY/FILEMAINT; FILE INPUT NAME ODESYSETUP/TD820
```

In either case, this is sufficient to enable the user to run ODESY initially.

2.6 FILE SIZES

The standard files supplied with the ODESY system have the following physical characteristics:

	Formats	Tank	System
Record length	60	180	45
Records per block	3	1	4
Records per area	300	200	480
Maximum areas	40	100	1

The system file is large enough for any user. However, some users may need to increase the file sizes of the Tank and Formats files. This may be done by modifying ODESY/FILEMAINT and then recreating the files, or it may be done by using DMPALL to copy the Tank and Formats files with larger file sizes.

For instance, to change the limits of the Tank file to 100 areas of 1000 records, Filemaint may be modified by entering from the SPO:

```
MO ODESY/FILEMAINT; FILE TANK BLOCKS.AREA=1000 AREAS = 100
```

The files must now be recreated.

Alternatively, the existing Tank file may be copied.
Enter from the SPO:

```
        EX DMPALL
```

Followed by:

```
<mix> AXPFM DSKDSK ODESYDATA/TANK ODESYDATA/TANK   180   1   1000   100
```

2.6 (Cont'd)

This will produce a new file ODESYDATA/TANK with the desired file limits.

When evaluating the file limits to be given to the Tank and Formats files, the following should be noted:

For the Tank file, each record entered by an operator occupies one record in the file. There will also be one extra record for each batch allocated, and a further 103 records for system use. Thus, the Tank file needs to be large enough to contain the following number of records:

103 + <number of batches> + <number of data records>

For the Formats file, there will be one record for each field, an extra record for each field with constants, and one record for each format header. There are also 799 records for system use. Thus, the Formats file needs to be large enough to contain the following number of records:

799 + <number of Formats> + <number of fields>
+ <number of fields with constants>

Note 1: The blocking factor for the tank file must not be changed, because the on-line version of the Extract program, and recovery system of the Edit depend upon the records being unblocked.

Note 2: If the file limits are increased by copying the files using DMPALL, then FILEMAINT should be modified at the same time, because squashing the Tank or Formats Files (see Section 3.11) or changing the Tank file record size (see Section 3.4) creates a new file using the file limits given to FILEMAINT.

2.7 RUNNING ODESY INITIALLY

1. Generate the Network Controller TEXT/HANDLER.

2. The following refers to the first station in the remote file STATIONS. Refer also to section 2.4 and the OPTIONS command in section 4.3.

   a. For a TD701, TD702, TD801 or TD802 terminal with braces as forms delimiters, the existing system configuration is adequate.

   b. For a TD701, TD702, TD801 or TD802 terminal with brackets as forms delimiters, enter from the SPO:

      MO ODESY/FORMAINT; SW1 = 2

      and:

      EX ODESY/FILEMAINT; FILE INPUT NAME ODESYSETUP/BRACKETS

   c. For a TD820, TD830 or TD730 terminal, enter from the SPO:

      MO ODESY/FORMAINT; SW1 = 5

      and:

      EX ODESY/FILEMAINT; FILE INPUT NAME ODESYSETUP/TD820

3. The standard system configuration may be printed by entering from the SPO:

   EX ODESY/FILEMAINT; FILE INPUT NAME ODESYSETUP/PRINT

4. The standard system formats, group name SYS, may be printed by running FORMAINT. Enter from the SPO:

    EX ODESY/FORMAINT

    The program opens the remote file STATIONS, and accesses the first station in the file

    Enter from this station:

    /HEL 1 SUP1

    followed by

    /PRI

    and the contents of the Formats file will be printed. If the SYS formats are ever removed, they may be reloaded by entering:

    /LOA ODESYSETUP/FORDUMP

5. The contents of the Tank file may be printed by running the PRINT Program. Enter from the SPO:

    EX ODESY/PRINT

    The report should show one batch, batch 999, which contains data for establishing the standard system configuration.

2.8 RECONFIGURATION OF THE ODESY SYSTEM

The contents of batch 999 matches the standard system configuration that the user has been supplied with. Reconfiguration is performed by changing the data in the batch, releasing it (using the EXTRACT Program), and then running FILEMAINT to make the required changes to the files.

The first record of the batch is a CONF maintenance request. If the user wishes, the maintenance request type may be changed to INIT, so that instead of modifying the existing ODESYDATA files, a new set of files is created. However if new files are created, the SYS formats and batch 999 will be removed. (The SYS formats may be reloaded by FORMAINT from the dump file ODESYSETUP/FORDUMP). It is recommended that the maintenance request type is left as CONF, and that batch 999 is retained, so that the system can be easily reconfigured in the future.

The reconfiguration process is as follows:

1. Enter from the SPO:

   EX ODESY/EDIT

2. The program opens the remote file STATIONS, and accesses the first station in the file.

   Enter from this station:

   /HEL 1 SUP1

   followed by:

   /OPE 999 ENT

3. Using the Enter, Modify, Insert and Delete commands, change the data in batch 999 to match the required system configuration, stations, operators, etc.

4. When all changes have been made enter:

   /REL

   followed by:

   /CLO

   and:

   /BYE

5. Enter from the SPO:

   ⟨mix no⟩ AX TER QC

   and the Edit and network controller will go to End-of-job.

6. Enter from the SPO:

   RE ODESYDATA/INPUT

   to remove the previous reconfiguration file (if it is still present).

7. Enter from the SPO:

   EX ODESY/EXTRACT

   and batch 999 will be extracted as ODESYDATA/INPUT

8. Enter from the SPO:

   EX ODESY/FILEMAINT; FILE INPUT NAME ODESYDATA/INPUT

9. The system has now been reconfigured.

On the first executions of FORMAINT, EDIT and EXTRACT after reconfiguration, the programs will dynamically change their memory requirements and go to End-of-job, after displaying PLEASE RE-EXECUTE. <u>This is not an error</u>.

3. THE FILEMAINT PROGRAM

3.1 GENERAL

FILEMAINT handles the creation and initialization of the Tank, Formats and System files. It also maintains the various records in the System file which the user may wish to alter. The following maintenance operations may be performed using FILEMAINT:

1. modifying system configuration parameters
2. inserting and deleting stations
3. inserting and deleting operators
4. modifying information and error texts or commands
5. inserting and deleting check digit verification tables
6. printing all or part of the system file
7. squashing the tank and formats files.

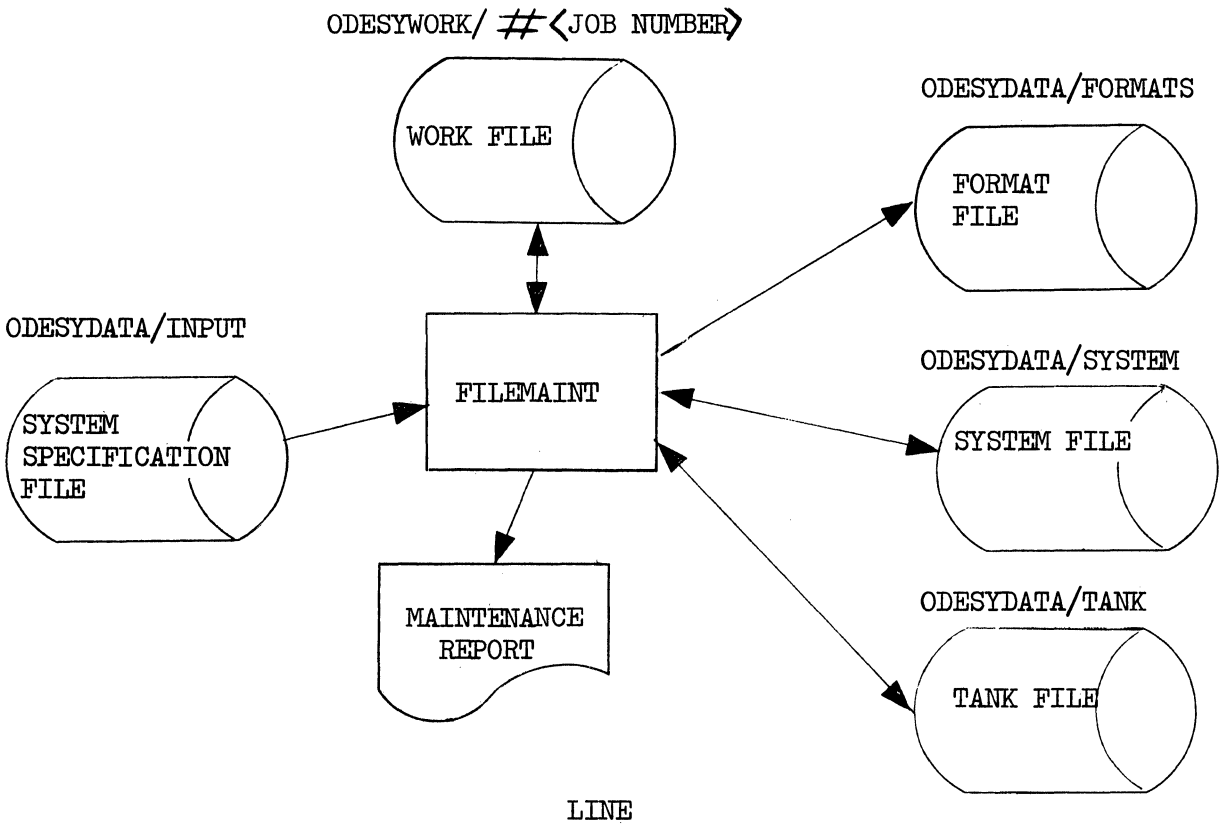
FILEMAINT does only a minimal amount of input parameter checking because it is assumed that all maintenance requests have been entered through (and checked by) the standard formats supplied with the system (group name = SYS).

However, these formats may only be used once the files have been initialized. In order to overcome this problem, a set of standard files called the STANDARD SYSTEM CONFIGURATION is provided to enable a minimal system to be run. This system can then generate a new file of initialization data which is specific to the user's requirements. This is explained in Installation Procedures, section 2.5.

When FILEMAINT is run, the individual maintenance requests are spooled to a work file (ODESYWORK/#<JOB NUMBER>) and a line is printed in the file maintenance report indicating the action requested. The disk directory is then searched to ensure that all necessary disk files are present. If they are present, and they are <u>not</u> in use, file maintenance is performed. If they are present and in use, then only print requests will be processed.

The record types on the input forms (Appendix C) are the names given to the standard formats mentioned above.

3.2 FILEMAINT BLOCK DIAGRAM



3.3 INITIALIZATION OF THE FILES

Initialization requires the same input forms as are used for configuration parameters. See section 3.4.

For the Tank and Format files, the files are created, various pointers are initialized and directories cleared, but no further action is taken. Maintenance of the Format file is performed by FORMAINT, and maintenance of the Tank file is performed by EDIT.

The System file is also created, and the configuration parameters are inserted. However, further maintenance requests for operators, stations and error texts are required before the System file is usable.

3.4 CONFIGURATION PARAMETERS

To set up the correct system configuration parameters input forms as per Appendix C1 should be completed. Note that the example given in Appendix C1 contains the standard Burroughs system configuration and even if it is only required to alter part of this, the whole record must be re-inserted. The details of the forms are as follows:

1. Maintenance Type - This should be a constant, either CONI, or if new files are to be created, INIT.

2. Number of characters to be allocated for storing constants in the EDIT - This should be a numeric value in the range 1 to 4095. If nothing is entered a default value of 200 is assumed.

3. Maximum number of formats (EDIT) - This defines the maximum no. of formats which are to be stored in the EDIT at one time. It should be a numeric value in the range 1 to 63. If nothing is entered a default value of 06 is assumed.

4. Maximum number of fields (EDIT) - This defines the maximum number of fields for all formats to be stored in the EDIT. It should be a numeric value in the range 1 to 2047. If nothing is entered, a default value of 100 is assumed.

5. Maximum user programs (EDIT) - This defines the maximum number of user programs which may communicate with the EDIT at one time. It should be a numeric value in the range 0-31. If nothing is entered a default value of 0 is assumed.

6. Input data size - This defines the maximum size, in characters, for the input data. It should be a numeric value in the range 01 - 1023. If nothing is entered a default value of 160 characters is assumed.

7. Record size for the Tank file - This should be at least 20 bytes larger than input data size above. Three record sizes are defined. Enter A for 180 bytes, enter B for 540 bytes or enter C for 1080 bytes. The default is A.

8. Page size for system reports - This entry defines the maximum number of lines per page to be printed in output reports. It is used by <u>all</u> programs. It should be a numeric value in the range 1 - 99. If nothing is entered a default value of 60 is assumed.

9. Date type - This entry is also relevant to output reports. Enter U for US standard date type - month, day, year (MM/DD/YY), enter I for international date type - year, month, day (YY/MM/DD), or enter E for European standard date type - day, month, year (DD/MM/YY).

10. Pack name - This is the name of the disk pack on which the network controller resides. Leave it blank if it is the system pack.

11. Family name - This is the name of the file family to which the network controller belongs. If there is no family name put the file name here.

12. File name - This is the file name of the network controller. It may be left blank (see above).

Sheet 2

1. Maintenance Type	- This is a constant, (CN2I)
2. Maximum number of characters of constant data per format (used by FORMAINT)	- This should be a numeric value in the range 100 - 4095. If nothing is entered a default of 500 is used.
3. Maximum number of characters of constant data per field (used by FORMAINT)	- This should be a numeric value in the range 25 - 1023. If nothing is entered a default of 100 is used.
4. Maximum number of fields per format (used by FORMAINT)	- This should be a numeric value in the range 63 - 255. If nothing is entered a default of 63 is used.
5. Maximum station buffer size (for FORMAINT)	- This applies to the terminal which will be used for Format Maintenance. Enter A for 300 (or 256) characters, Enter B for 1024 (or 960) characters, enter C for 2000 (or 1920) characters, enter D for 4000 (or 3840) characters.
6. Maximum number of formats (EXTRACT)	- This defines the maximum number of formats to be stored in EXTRACT at one time. It should be a numeric value in the range 1 - 31. If nothing is entered a default of 10 is used.
7. Maximum number of fields (EXTRACT)	- This defines the maximum number of fields for all formats to be stored in EXTRACT at one time. It should be a numeric value in the range 1 - 1023. If nothing is entered a default of 200 is used.
8. Maximum output record size (used by EXTRACT)	- This is the maximum record size for an output file. It should be a numeric value in the range 80 - 999 bytes. Default is 400.

Sheet 3

1. Maintenance Type - This is a constant (CN3I)

2. Duplicate anything option - This controls how the field option Duplicate In will work. Enter Y for duplicate anything, or enter N for restricted duplication.

3. Restrict options - For the full list of restrict options, see appendix C1, sheet 3. In each case enter R for restricted to supervisors, or enter N for not restricted.

4. Include options - For the full list of include options, see appendix C1, sheet 3. In each case enter I for include or E for exclude.

3.5 STATION PARAMETERS

To insert a new station into the System file or to delete an existing station, an input form as per Appendix C2 should be completed. The details of the form are as follows:

1. Maintenance type - This is a constant (STA).
2. Maintenance action - This should be either I for inserting a station, or D for deleting a station.
3. Logical station number - This is taken from the NDL listing and will be a numeric value in the range 1 - 99.
4. Station name - This should be the name that appears in the NDL listing.
5. Station buffer size - This is the physical size in characters of the terminal's buffer and will be a numeric value in the range 10 - 4095. The default value is 1920 characters.
6. Forms set - This defines the forms delimiters used for the station. Enter either A for braces "{ }", B for square brackets "[ ]" or C for US and RS. If nothing is entered a default value of A = braces is assumed.
7. Control status - This should be C for a permanent control station, otherwise leave blank.

Note: It is not necessary for the logical station numbers to be in sequence, but they must agree with the NDL specification.

3.6 OPERATOR PARAMETERS

To insert a new operator or to delete an existing operator an input form as per Appendix C3 should be completed. The details of the form are as follows:

1. Maintenance type - This is a constant (OPR).

2. Maintenance action - This should be either I for inserting an operator or D for deleting an operator.

3. Operator number - This should be a numeric value in the range 1 - 99. Note that operator numbers do not have to be consecutive, they may be assigned in any order.

4. Operator name - This is alphanumeric and should not be more than 16 characters in length.

5. Operator password - This is alphanumeric and should not be more than four characters in length. The first character must be alphabetic.

6. Operator security level - This should be a numeric value in the range 1 - 99. The relationship between operator and format security level is described under 'SECURITY' (Section 1.7).

7. Operator type - This should be either O for an operator or S for a supervisor. See SECURITY (Section 1.7)for the functions of Operator and Supervisor . The default value is O.

3.7 ERROR TEXTS

Each error message number used by EDIT or FORMAINT has a unique meaning associated with it. This means that the actual text of the error message may be modified or translated into another language but the message number must always remain the same. For a given error situation the system will always output the same message number.

To modify error message texts an input form as per Appendix C4 should be completed. The details of the form are as follows:

1. Maintenance type	-	This is a constant (TXTI).
2. Text number	-	This must be the same as that allocated to the text to be modified. It will be a numeric value in the range 1 - 199. EDIT messages are numbered 1 - 100 and FORMAINT messages 101 - 199.
3. Message text	-	Enter the replacement text required. Maximum length for an error message text is 31 characters including embedded blanks.

The standard error message texts currently implemented may be found in Appendix F.

3.8 COMMANDS

Each command used by the Edit and FORMAINT is identified by its first three characters. In order to overcome local language difficulties, it is possible to modify these three-character abbreviations. Also included here are certain parameters which are used with the commands.

To modify commands and parameters for the Edit, input forms as per Appendix C5 and C6 should be completed. To modify commands and parameters for FORMAINT, an input form as per Appendix C7 should be completed. The details of the forms are as follows:-

1. Maintenance type	-	This is a constant (CMDI)
2. Command set number	-	This should be 1 for the first 20 Edit commands, 2 for the remaining Edit commands, or 3 for the FORMAINT commands.
3. Commands and parameters	-	Three characters for each. The default values are shown.

For a full list of the commands and parameters, please refer to Appendix F, sections 4 and 5.

Note: If the user wishes to modify only some of the commands, he or she must remember to tab to the end of the format template when it is displayed by the Edit. If for example CMD2 is being displayed to change the parameters QC and ALL only, all the other commands or parameters in this format must be tabbed over, not just ON, OFF and NOW. The standard commands appear as default values on all templates).

3.9 CHECK DIGIT VERIFICATION SCHEMES

To insert a new check digit scheme or to delete an existing check digit scheme an input form as per appendix C8 should be completed. The details of the form are as follows:-

1.	Maintenance type	- This is a constant (CDV).
2.	Maintenance action	- This should be either I for inserting a new scheme or D for deleting an existing scheme.
3.	Scheme number	- This should be a numeric value in the range 1 - 9.
4.	Modulus	- This should be a numeric value in the range 10 - 15.
5.	Remainder	- This should be a numeric value in the range 0 - 15. The default value is 0.
6.	Calculation type	- This should be E if the products are entered, A if the products are calculated by addition from the weights or M if the products are calculated from the weights by using the modulus.
7.	Complemented	- This should be Y if the check digit is to be complemented or N if not. The default value is N.
8.	Number of weights	- This should be a numeric value in the range 1 - 10.
9.	Weights or products	- For calculation type M or A the hexadecimal equivalent of the weights should be entered in this field. For calculation type E, enter the hexadecimal equivalent of the products.

For a full explanation of how individual check digit schemes operate, please refer to Appendix E.

3.10 PRINT FACILITIES

To print either the entire file or selected record types in the file an input form as per Appendix C9 should be completed. The details of the form are as follows:-

1. Maintenance type - This is a constant (PRT).
2. Print requests - The valid record types are:-

   CON for configuration data

   CTL for control data

   STA for station data

   OPR for operator data

   TXT for error text data

   CMD for command data

   CDV for check digit data

   ALL for the complete System file

3.11 SQUASH FACILITIES

The tank file expands in size as data is entered, but it does not automatically contract in size. Operators may delete batches, but this simply creates free records within the existing tank file, it does not reduce the size of the file. As new data is entered, these free records are used before any records are added to the end of the file. However, if the tank file has already grown too large, the solution is to squash the file, which moves all free records to the end, and reallocates the space.

Similarly, deleting a format or group from the formats file creates free records, and the file must be squashed if it ever grows too large.

To squash either the tank file or the formats file or both, an input form as per Appendix C10 should be completed. The details of the form are as follows:

1. Maintenance type - This is a constant (SQU).
2. Squash requests - The valid types are:
   TNK for tank file
   FOR for formats file

4. THE FORMAT MAINTENANCE PROGRAM (FORMAINT)

4.1 GENERAL

During data entry, the terminal operator works with a batch, and this batch will eventually be output as a file. A GROUP HEADER specifies the file characteristics of that file, and each batch is associated with one and only one GROUP.

A FORMAT is a complete specification of one screenful of data. It specifies the screen layout that the operator sees, with field descriptions and forms delimiters; it specifies the audit checks to be applied to the data the operator keys in; it specifies how that data is to be processed, and where it will appear in the output file; and it specifies where that data is to be printed in the journal, produced at the time the file is output (Note that the journal is optional).

Each group will probably contain several formats, since the file produced usually has more than one record type.

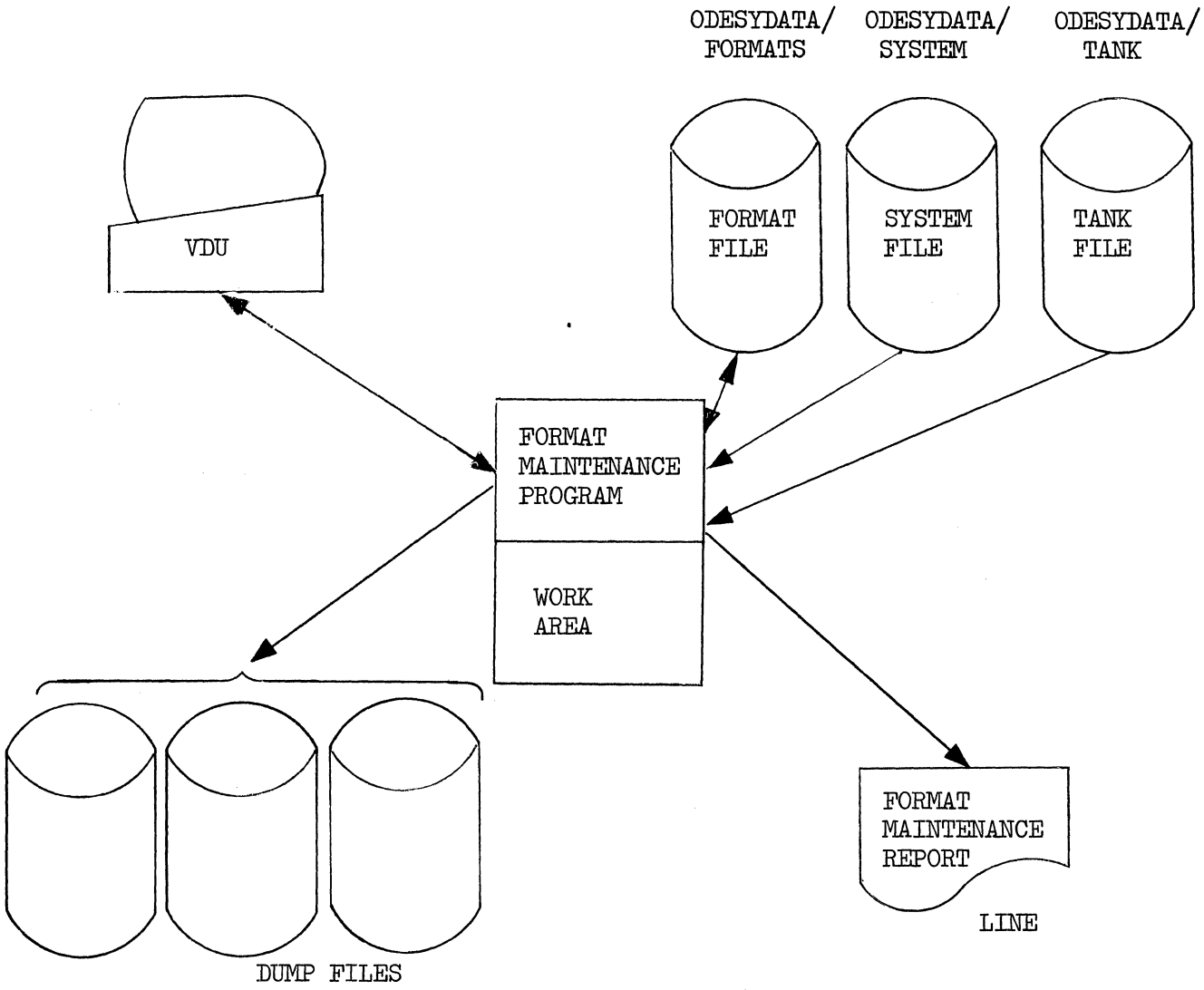
The format contains a FORMAT HEADER and up to 63 FIELDS. Each field contains FIELD DETAILS, and optionally CONSTANT DETAILS. The constants are required for range checks, value checks and inserted constants.

The groups and formats are stored in the FORMAT FILE, and the purpose of this chapter is to describe the use of the FORMAT MAINTENANCE PROGRAM.

The Format Maintenance Program is an on-line program, but unlike the Edit it communicates with only one terminal during each format maintenance session. The terminal operator keys in both commands and data. All commands are preceded by the "slash" character: /. Only the first three characters of each command are significant, for example /INS has the same meaning as /INSERT. The key words GROUP, FORMAT and FIELD may similarly be truncated to GRO, FOR and FIE. Any input not preceded by "slash" or "question mark" is interpreted as data. Input preceded by the "question mark" character: ? is interpreted as control information, and is zipped to the MCP.

The commands are described in section 4.3, and the meaning of the data in sections 4.4 - 4.8. Section 4.9 contains a simple worked example, which will give the user an opportunity to enter a simple format and group from a terminal. Section 4.10 shows some more complex examples, which should be of particular interest to users of the 256-character TD701 terminal, or the 1920-character TD802 terminal. Copies of the blank layout forms and a list of commands appear in the Appendices.

4.2 FORMAT MAINTENANCE PROGRAM BLOCK DIAGRAM



4.3 FORMAT MAINTENANCE COMMANDS

AMEND, DELETE, INSERT, LIST

These commands control the Format Work Area. The work area is able to contain one group header, one format header and up to 63 fields at any one time. The data in the work area may be input from the terminal, using the /INSERT command, or it may be obtained from the format file, using the /GET command. Once the data is in the work area, it may be listed, amended or deleted, using the appropriate command. If the data in the work area is a new group and/or format, or if changes have been made to an existing group or format, then the format file may be updated using the /SAVE command. The /GET and /SAVE commands are mentioned in more detail separately.

The parameters associated with each of these four commands (/INSERT, /LIST, /AMEND and /DELETE) are the same, and they indicate whether the command applies to the group header (parameter = GROUP or blank), the format header (parameter = FORMAT), or one of the fields (parameters = the field sequence number, or FIELD followed by the field sequence number).

Format information is always entered with the terminal in FORMS MODE. In each case, the operator is presented with a TEMPLATE consisting of a top line of control information, followed by a set of descriptions, with space for format data to be entered. For each item of format data, an "unprotected area" appears on the screen, enclosed within forms delimiters.

The cursor will only move within the unprotected areas, so that the operator cannot overwrite the descriptions or the control information, which are protected data. The forms delimiters on the TD700/TD800 range of terminals are either braces or square brackets.

Each description refers to one unprotected area, and indicates what should be entered. With a little experience, the descriptions should provide enough guidance to an operator entering formats to make it unnecessary to refer to this chapter.

The insertion of a complete format is divided into three stages: entry of the group header, entry of the format header, and entry of the fields. The group header is entered using two templates. The format header is entered using one template. For each field, there will be at least two templates. If an inserted constant or a range check is required for the field, there will be a third template for the entry of the constants. If a value check is required for the field, the third template reappears if there are more than five constants to enter.

Note that the format header cannot be inserted if there is no group header present, and similarly a field may not be inserted if there is no format header present.

The worked example in section 4.9 shows how a simple format with three fields may be entered.

During insertion, it is anticipated that the user will normally enter group header, format header, field, field, field, etc. Therefore the second group template is automatically followed by the format template, which is automatically followed by the first field template; and each set of field/constant templates is automatically followed by the first field template once more. To break out of this cycle, the user enters a new command.

The list command also anticipates that the user will wish to examine the work area in the order group header, format header, fields. In this case the next template plus data is obtained by transmitting a single space.

The amend command combines list and insert. The existing data for the group header (or format header or field) is listed, and the user makes changes and transmits the whole of the data (not just the changes), as if the group header were being inserted.

The delete command permits erroneous data to be deleted from the work area. However note that /DELete FORmat deletes the format header and all fields, and similarly /DELete GROup deletes the whole Work Area.

For the insert and amend commands, the data being input is checked for validity. If there are no errors, the next template is displayed. However if errors are found, the template is redisplayed together with only the data in error. The user now has two choices. Either the whole of the data may be rekeyed and transmitted or a single "slash" may be transmitted, whereupon

the template is redisplayed with the data originally entered, so that the erroneous data may be corrected. IT IS NOT POSSIBLE TO COMPLETE THE ENTRY OF A GROUP HEADER, FORMAT HEADER OR FIELD WHICH CONTAINS ERRORS.

The checks applied to the data are described in detail in sections 4.4 - 4.8.

BYE, TERMINATE

At the end of a format maintenance session, the user should input one of these commands.
/BYE will close down the format maintenance program only.
/TERMINATE will close down both the format maintenance program and the network controller.

DELETE - refer to AMEND

DISPLAY

The format in the work area is displayed at the terminal as it would appear during Data Entry. If the user does not like the screen layout, the amend command can be used to modify the format as necessary before saving it.

DUMP, LOAD

Formats may be stored in what are known as dump files. The load command

allows them to be loaded from a dump file to the Format file, and the dump command allows formats to be dumped from the Format file to a dump file. The user will probably need to know about the load command in order to obtain certain standard formats after the Format file has been initialised.

The parameters for both commands are the same. The first parameter is the dump file name. The pack name (or pack-id), the family name (or multi-file-id) and the file name (or file-id) should be separated by "slashes", and no spaces should appear in the name as a whole. The other parameters are optional, and consist of a group name, or a group name and a format name.

For the load command, the whole of the dump file can be loaded by specifying only the file name. Alternatively, a specific group can be loaded by naming that group, or a specific format with its group header can be loaded by naming the group and format. If a group or format of the same name already exists in the Format file, it will be overwritten by the data in the dump file.

For the dump command, the whole of the Format file can be dumped by specifying only the dump file name. Alternatively, a specific group can be dumped by naming that group, or a specific format and its group header can be dumped by naming that group and format. If the specified dump file already exists on disk, data will be added to the end of that file, otherwise a new disk file will be created.

GET

This command is used to obtain formats from the Format file and place them in the work area, where they may be changed by the user.

The first parameter is a group name, and the optional second parameter is a format name. If the format name is omitted, only the group header will be placed in the work area, otherwise both the group header and the format will be placed in the work area.

HELLO

This must be the first command input at the start of a format maintenance session. The parameters required are an operator number and password. The number and password are compared with the data in the System File, and if the values do not agree the user will not be able to continue.

INSERT - refer to AMEND

LIST - refer to AMEND

LOAD - refer to DUMP

OPTIONS

By default the format maintenance program expects the terminal to be a TD701 with braces as forms delimiters. If the forms delimiters for the terminal are

square brackets, or the terminal is a TD820, TD830 or TD730, then it is necessary to change the setting of external switch SW1, or to use the options command immediately after the hello command. If the terminal is a TD801 or a TD802, the options command can be used to take advantage of the larger screen. The parameter is:

1 for TD700 + braces
2 for TD700 + brackets,
3 for TD800 + braces,
4 for TD800 + brackets,
5 for TD730
6 for TD820, TD830

External switch SW1 can be changed by entering from the SPO:

MO ODESY/FORMAINT; SW1 = ⟨terminal type⟩

where ⟨terminal type⟩ is between 1 and 6.

PD (print directory)

This command is used to obtain a directory of all group names in the format file, or of all format names associated with a specific group name.

The PD command with no parameters gives a directory of group names. The PD command followed by a group name gives a directory of format names for that group.

If there are too many names for one screen, then the first "page" of the directory is displayed in response to the PD command, and further "pages"

may be obtained by transmitting a single space.

PRINT

The print command is used to produce a formatted line printer report of the data in the format file or the work area. It should be noted that reports are not automatically produced when a format is /SAVed, so that the user must explicitly key in a print command.

The whole of the Format file can be printed by keying in the print command with no parameters. Alternatively a specific group can be printed by giving the group name as a parameter, or a specific format (with its group header) can be printed by giving the group and format names as parameters.

The work area can be printed by inputting the key words WORK AREA (group name = WORK, format name = AREA), and for this reason the user should avoid giving these names to an actual group and format.

REMOVE

This command is used to remove a specific group or format from the Format file.

The first parameter is a group name, and the optional second parameter is a format name. If the format name is omitted, the whole group will be removed from the Format file, otherwise only the named format will be removed.

SAVE

This command is used to update the Format file with the contents of the work area. However, the contents of the work area are not destroyed by the save command. They remain intact until the user inputs an explicit delete (group) command, or performs an implicit delete by using the get or insert (group) commands.

SS

This command is used to send a message to the Central Machine Console (or SPO).

An operator at the Central Machine Console may send a message to the terminal by entering ⟨mix number⟩ AX ⟨text⟩.

TERMINATE - refer to BYE

/

A single "slash" serves as a repeat command. It is used in conjunction with the amend, insert and list commands. Refer to AMEND.

? ⟨control information⟩

Any input preceded by a "question mark" is interpreted as control information, and is zipped to the MCP.

<data>

Any input not preceded by "slash" or "question mark" is interpreted as data.

Refer to sections 4.4 - 4.8.

4.4 ENTRY OF FORMAT DATA

Format data may be divided into group header details, format header details, field details and constant details, and each is discussed in a separate section.

When the appropriate template appears on the screen in response to the insert (or amend) command, the description preceding each unprotected area indicates what may be entered. For those options where mnemonics are used, the description indicates the choice. For instance, the description preceding the Justify In option is

JUST.IN: ,L,R

so that the user may enter L or R, or leave it blank. For the other options, an alphanumeric value or a numeric value or an identifier is required, and the definitions of these terms are now given.

For the purposes of this program, a value is alphanumeric if it conforms to the following rules:

1. First character is alphabetic (A-Z).
2. The remaining characters are alphabetic (A-Z), numeric (0-9), or blanks.
3. The value is left justified.
4. If blanks appear, they must be contiguous at the right hand end.

For the purposes of this program, a value is numeric if it conforms to the following rules:

1. The only allowable characters are numeric (0-9), and blanks.
2. The numeric characters must be contiguous.
3. Blanks may appear before the numeric part, and/or after the numeric part.

For the purposes of this program, an identifier conforms to the following rules:

1. The allowable characters are alphabetic (A-Z), numeric (0-9), special ($ # & * . -) or blank.
2. The characters, including the six special characters, may occur in any combination.
3. The value is left justified.
4. If blanks appear, they must be contiguous at the right hand end.

4.5 GROUP HEADER DETAILS

Group Name

This name is used to identify the group. It should be a non-blank alphanumeric value. If the group name is left blank, the error is indicated by four asterisks.

File Name

This is the file name associated with the group. It contains three fields of ten characters. These are the pack name (or pack-id), the family name (or multi-file-id) and the file name (or file-id) respectively. For tape files, the pack name should be left blank.

Only the family name is mandatory, and if it is left blank, the error is indicated by ten asterisks. Each name should be an identifier.

Record Size

The record size for the file should be a numeric value in the range 1 - 999. If the field is left blank, a default value of 180 is used.

Blocking Factor

The blocking factor for the file should be a numeric value in the range 1 - 99. If the field is left blank, a default value of 1 is used.

Records per Area

This is only applicable to disk files. The number of records per area for the file should be a numeric value in the range 1 - 9999. If the field is left blank, a default value of 500 is used for disk files.

Number of Areas

This is only applicable to disk files. The number of areas for the file should be a numeric value in the range 1 - 99. If the field is left blank, a default value of 40 is used for disk files.

Report

If a journal is required for batches using this group, this option should be set to R, otherwise leave blank. The journal is produced during extraction.

It should be noted that the Print Column option (see Field Details, subsection 7)

controls the layout of the journal, but the journal will not be produced at all if the Report option is left blank.

Medium

At present only disk and tape are implemented. Disk includes disk cartridge, disk pack and head-per-track disk. Tape includes 9-track tape, 7-track tape, PE-tape and NRZ-tape. If tape is required, this option should be set to T, and the pack name should be blank. If disk is required, leave blank.

Recording Mode

Not yet implemented.
This is only applicable to tape files. The option should be set to A for ASCII, or to B for BCL. If EBCDIC is required, leave blank.

Label Type

This is only applicable to tape files. The option should be set to A for ANSI tape labels, or U for unlabelled. If Burroughs standard labels are required, leave blank.

Report Input Record Number

If a journal is to be produced, then this option may be used to obtain the input record number on every line in the journal. (The input record number is the record number in the batch). The record number printed will occupy six columns, and the numeric value entered for this option specifies the starting print column position.

If there is no journal, or the input record number is not to be printed, then leave blank, otherwise enter a numeric value in the range 1 to 127.

Report Output Record Number

If a journal is to be produced, then this option may be used to obtain the output record number on every line of the journal. (The output record number is the record number in the output file, or application file). The record number printed will occupy six columns, and the numeric value entered for this option specifies the starting print column position.

If there is no journal, or the output record number is not to be printed, then leave blank, otherwise enter a numeric value in the range 1 to 127.

4.6 FORMAT HEADER DETAILS

Format Name

This name is used to identify the format within the group. It should be a non-blank alphanumeric value. If the format name is left blank, the error is indicated by four asterisks.

It should be noted that duplicate format names are allowed, providing they are in different groups. Thus "G1 FORM" and "G2 FORM" are two distinct formats in groups G1 and G2, but they have the same format name FORM.

Next Format

This is the name of the format which will automatically appear on the screen during Data Entry after successful entry of data for the current format. It may be the same as the format name, in which case a format will continuously follow itself until another Data Entry command is entered.

This option should be an alphanumeric value, or blank.

The entry of a batch of data is very efficient where a header record format is automatically followed by a detail record format, which automatically follows itself until the operator keys in /CLOSE (or keys in the name of a trailer record format).

Security

A security level between 1 and 99 is given to each format and to each operator. An operator is not allowed to enter data if the security level of the format is greater than the security level of the operator.

The security should be a numeric value in the range 1 to 99. If the field is left blank, a default value of 50 is used.

Verify

This option specifies the type of verification allowed for records entered under the control of this format.

For a No-Verify format, verification is not permitted at all. For a Display-Verify format, the data originally entered is redisplayed, and the operator may visually check the data, amend it if necessary, and retransmit it. For a Standard-Verify format, the data is not redisplayed, and the operator must rekey the whole of the data. The data transmitted will be checked for comparison errors.

The option should be set to N for No-Verify, or to D for Display-Verify. If Standard-Verify is required, leave blank.

Format Type

This option specifies the type of the format, for instance data entry format, batch header format, etc.

At the moment, there are six format types, of which four are used for the inquiry facilities referred to in section 10. The simple format types are data entry formats and batch header formats. The inquiry format types are key data formats, supply data formats, check data formats and check-and-supply formats. Inquiry formats require a user program.

The option should be left blank for a data entry format, or set to B for a batch header format, or to K for a key data format, or to S for a supply data format, or to C for a check data format, or to U for a check-and-supply format.

User Program Name

This is only applicable to inquiry formats. For a key data format or a supply data format or a check data format, or a check-and-supply format, a user program name must be entered.

It contains three fields of ten characters. These are the pack name, the family name and the file name respectively. Only the family name is mandatory, and if it is left blank, the error is indicated by ten asterisks. Each name should be an identifier.

4.7 FIELD DETAILS

Field Sequence Number

This number is used to identify the field within the format. It should be a numeric value in the range 1 - 999. If the sequence number is left blank, the error is indicated by three zeroes.

It is good practice to number the fields in tens (that is 10, 20, 30, etc), so that should it be necessary to insert an extra field at a later date, then it is easy to do so (for instance, to insert field 15, which will be placed between 10 and 20).

Field Name

During Data Entry, this name is displayed to the left of the unprotected area for the field. The name is truncated by omitting any spaces at the right hand end, and if the name is left blank entirely, only the unprotected area, enclosed by forms delimiters, will appear.

For this option, up to ten characters may be entered. The name may consist of any combination of alphabetic, numeric or special characters.

Field Length

This specifies the number of character positions on the screen which will be allocated for this field. It should be a numeric value.

For alphanumeric fields, the limit is 160 characters, except for those fields where inserted constants or value check constants are required, where the limit is 28 characters. For numeric fields the limit is 16 characters (15 digits plus a sign). This limit also applies to fields with range check constants, which must be numeric. If the appropriate limit is exceeded, or the field length is left blank, an error will be indicated. A field length of zero is not permitted.

If a field length of 16 is specified for a numeric field, the Sign option must be set to either + or - (see Sign option).

Field Type

Only two field types are defined, alphanumeric and numeric. Data for numeric fields may contain only the characters 0 - 9 with an optional leading sign (+ or -). There are no restrictions on the data for alphanumeric fields. Decimal fields should be defined as type numeric. The decimal point is not a permitted character, but alignment on the decimal point may be arranged by using the Justify In or Justify Out options, or by defining two separate fields.

Duplicate In

If the Duplicate option is set, the data for the field is duplicated (on the screen) from the previous record, provided that the record was entered using the same format. However, the duplicated value is still enclosed in forms delimiters, so that the operator is able to change the value at any time.

The option should be set to D for duplication, otherwise leave blank.

End Line

This option allows the screen layout to be controlled more easily. If it is set, a "carriage return" character is inserted after the right hand forms delimiter for the field, so that the next field begins on a new line. If required, the option should be set to E.

Inserted Constant, Range Check and Value Check

These are dealt with in more detail under Constant Details, Section 8. The length of the constant is the same as the length of the field for which it is specified.

The option should be set to C for an inserted constant, or R for a range check, or V for a value check. Otherwise leave blank.

Crossfoot Check

This is only applicable to numeric fields. A crossfoot register exists, to which (numeric) values may be added, or from which they may be subtracted. It provides a subtotal check for the record as a whole. An error is indicated if the crossfoot register is not zero at the end of checking the record.

The option should be set to + for "add to crossfoot register", or to - for "subtract from crossfoot register". Otherwise leave blank.

Check Digit Verification

This is only applicable to numeric fields. Up to nine check digit verification (CDV) schemes may be defined, and in order to check the value entered for this field against a particular scheme, the identifying number of the scheme must be specified here. There is no check that a CDV scheme number actually exists on the system file. The user must ensure that this is so.

The option should be set to a number in the range 1 - 9 if check digit verification is required, otherwise leave blank. See also Appendix E.

Batch Total

This is only applicable to numeric fields. Five batch total registers exist. If the value entered for this field is to be added to a particular batch total register, the number of the register should be specified here. Batch totals are checked immediately before the batch is closed. For a batch header format, this option indicates which batch total the field is meant to be.

The option should be set to a number in the range 1 - 5 if this field either contributes to a batch total or is a batch total, otherwise leave blank.

Mandatory

A field is mandatory if its value is not allowed to be blank.

The option should be set to M if this is a mandatory field, otherwise leave blank.

Justify In

If a field must be entered with left or right justification, this option may be used. It should also be noted that fields may be output with left or right justification by using the Justify Out option.

The option should be set to L for left justify check, or to R for right justify check. Otherwise leave blank.

Output Position

This specifies the position that the value entered for the field is to occupy in the output record. It should be a numeric value. Units are bytes (= 8 bits), and the first byte in the output record is denoted as byte 1 (one). The limit is the record length specified in the group header details.

If the field is not to be output (for instance a value that is only to appear in the journal), then leave blank.

Write After

This option controls the writing of records to the output file by the Extraction Program, and proper use of it will enable multiple output records to be produced from one input record, or one output record from several input records. Worked Example 2, Section 4.10 shows how this can be done.

If the user's requirements are for each input record (or screenful of data) to produce one output record, then all that is necessary is to set this

option to Write-And-Clear for the last field of each format. However, if greater complexity is required, then the following two paragraphs should be studied carefully together with Worked Example 2.

The Write After option can take values of Write-And-Clear, Write-Only, Save and blank (blank implies no action). It is also necessary to know that there are two buffers, an output buffer and a duplication buffer. During the extraction of each input record, the data is processed field by field, in field sequence number order. The data for a field is taken from the input record, it is justified, packed and a sign inserted if the relevant field options are set, and it is placed in the output buffer ready for writing to the output file. It will also be placed in the duplication buffer if the Duplicate Out option is set. The Write After option for the field is then examined. If the option is set to Write-And-Clear, the contents of the output buffer are written (provided the input since the last write was not all blank), and both the output buffer and the duplication buffer are cleared. If the option is set to Write-Only, the buffer is written (provided the input since the last write was not all blank), but the contents of the duplication buffer are loaded into the output buffer. Write-Only permits several output records with information in common to be produced from one input record. The common data is normally contained in the first few fields, which will have the Duplicate Out option set, followed by the fields for the information specific to the first record to be written. The Write After option is set to Write-Only for the last of these fields. The information specific to the second record then follows, with output positions such that it does not overwrite the common data. For an example see format MULTIOUT in Worked Example 2.

The qualification that the buffer is not written if the input since the last write was all blank prevents redundant output records being written. It may happen, at the end of a batch for instance, that it is desired to produce only one record, even though the format is capable of producing several. In this case the terminal operator keys in the common data and the details specific to the first record, and leaves blank the remainder. An extension of this, which requires the use of Save, is illustrated by the format MULTIALT in Worked Example 2. This format is capable of producing two output records. It contains common data, but the detail information is completely different for the first and second records. In order to use the format to produce only the second record, it is necessary to set the Write After option to Save for the last field of the common data. The effect of Save is to treat as blank the data contained in the preceding fields. Thus if the data specific to the first record is blank the record will not be written, even though the common data was not blank.

The option should be set to S for Save, or to W for Write-Only, or to C for Write-And-Clear. Otherwise leave blank.

Duplicate Out

This should be used in conjunction with the Write After and Print After options. If this option is set, the data for the field is placed in the duplication buffers as well as being placed in the output buffers. Setting the Write After option to Write-And-Clear will clear the duplication output

buffer, and setting the Print After option to Print-And-Clear will clear the duplication printer buffer.

The option should be set to D for duplication, otherwise leave blank.

Blank Fill

This is only applicable to numeric fields. The default is zero fill, that is blanks are replaced by zeroes. If the field is signed, the sign will be moved either to the most significant digit position, or to the least significant digit position, depending on the sign convention.

The option should be set to B for blank fill of numeric fields, otherwise leave blank.

Packed

This is only applicable to numeric fields. The use of this option enables numeric fields to be converted to 4-bit packed (hexadecimal) storage for output. Because the Output Position option is specified in units of 8 bits, the Packed option has been extended to enable a packed field to start either at a byte boundary (as usual) or 4 bits later at a "Half-byte Boundary".

The option should be set to P for packed output, or to H for packed output with "Half-byte" start position. Otherwise leave blank.

Justify Out

Fields may be justified for output using this option. For instance, numeric fields may be entered left justified, and then right justified during extraction.

The option should be set to L for left justification, or to R for right justification. Otherwise leave blank.

For numeric fields it is recommended that either the Justify In option is set to R, or the Justify Out option is set to R, in order to achieve correct alignment on the (assumed) decimal point. No provision is made for the insertion of a decimal point in numeric fields.

Sign

This is only applicable to numeric fields. If signs are allowed, this option must be set, and the value set serves as a default value of the sign. If a sign is present in the value entered for this field during Data Entry, that sign will be used, otherwise the default sign will be inserted.

The option should be set to + for plus sign insertion, or to - for minus sign insertion. Otherwise leave blank.

It was noted earlier that numeric fields are allowed to contain a maximum of

15 digits plus a sign, so that a numeric field of length 16 must have the Sign option set.

Sign Convention

This is only applicable to numeric fields with the Sign option set. The option denotes the position of the sign in the output record. The default is the most significant position (left most), but the user may particularly require the sign to be in the least significant position (right most). For unpacked numeric fields, an overpunch may be requested.

The option should be set to 1 for least significant position, or to 2 for most significant position overpunch, or to 3 for least significant position overpunch. Otherwise leave blank.

It should be noted that a signed, unpacked, numeric field of length L with an overpunch will produce a field of length (L - 1) bytes in the output record. In all other cases a field with an input length of L will produce a field with an output length of L, that is L digits (= L/2 bytes) for packed fields, otherwise L bytes. The Field Length option is the input length.

Print Column

This specifies the position that the value entered for the field is to occupy in the journal. It should be a numeric value. The first column in the journal is denoted as column 1 (one), and the limit is 132.

If the field is not to be printed, then leave blank.

It should be noted that a journal will not be produced unless the Report option in the group header is set to R.

Print After

This option controls the printing of lines of the journal by the Extraction Program. It serves the same purpose in relation to the journal as the Write After option serves in relation to the output file, and it has similar settings of Save, Print-Only and Print-And-Clear. If the user's requirements are for each input record (or screenful of data) to produce one line in the journal, then all that is necessary is to set this option to Print-And-Clear for the last field of each format. Otherwise the user should refer to Worked Example 2.

The option should be set to S for Save, or to P for Print-Only, or to C for Print-And-Clear. Otherwise leave blank.

Print Name

This option enables the field name to be printed in the journal as well as the field data. If this option is set, the field name will be printed, starting in the column indicated by the Print Column option. The name will be truncated by omitting any spaces at the right hand end, and one space will separate the right hand end of the truncated field name from the field data. For example, a field of length six, with a field name "TOTAL" that is to be printed, will occupy 5 + 1 + 6 = 12 columns in the journal.

The option should be set to P if the name is to be printed, otherwise leave blank.

4.8 CONSTANT DETAILS

During the entry of format information, a constant template will appear on the screen if the constant option is set to C, R or V. The template will allow the entry of one (for inserted constant), three (for range check) or five (for value check) constants, and the length of each constant will be the same as specified in the Field Length option. The Value Check template will reappear to allow further constants to be entered if the fifth constant is not blank.

On the Field Layout form in Appendix C13, the lower part of the form is set aside for constant details. The recommended way of filling in the form is to enter the field sequence number in the first three columns, and the constant value or values in the remainder. The field sequence number is then a cross reference to the corresponding field details above.

Inserted Constant

During Data Entry an inserted constant will appear on the screen enclosed in forms delimiters. Therefore the constant serves as a default value rather than an enforced insertion, since the operator is able to change the value. For this reason, the user may wish to include checks on fields with inserted constants.

If the Constant option is set to C, the constant template which appears will allow a constant of the required length to be keyed in. The data will be checked, and non-numeric data for a numeric field will result in an error, as will incorrectly justified data for a field with the Justify In option set. A constant of blanks is not allowed, since a field with a default value of blanks could be equally well described as a non-mandatory field.

Range Check Constants

A range check is only applicable to numeric fields. Three constants are required, the first being the default value which is displayed during Data Entry (same as Inserted Constant), the second being the lower limit, the third being the upper limit. During Data Entry, the value entered for the field will be checked to ensure that it lies within the limits.

If the Sign option is set, the sign will be taken into account during the range check.

If the Constant option is set to R, the constant template which appears will allow three constants of the required length to be keyed in. The data will be checked, and non-numeric data will result in an error. If the Sign option is set, then the lower and upper limits are allowed to be signed. The first constant, the default value, may be blank.

Value Check Constants

If a value check is specified for a field, the value entered for the field during Data Entry will be checked to ensure that it corresponds with one of the value check constants. The first constant will also be used as a default value, and will be displayed during Data Entry (same as Inserted Constant).

If the Constant option is set to V, the constant template which appears will allow up to five constants of the required length to be keyed in. The data will be checked, and non-numeric data for a numeric field will result in an error. If five constants are entered, the constant template will reappear to allow further constants to be entered if necessary. The first constant, the default value, may be blank.

4.9 Worked Example 1

Start the Format Maintenance Program

Key in

/HEL 1 SUP1

Transmit.

The reply is

START OF SESSION

Key in

/INS

Transmit.

The reply is group template 1.

Key in

group name:	G1	tab
file name:	tab	G1 FILE

Transmit.

The reply is group template 2.

Leave the unprotected areas blank.

Transmit.

The reply is format template.

Key in

format name:	FMT1

Transmit.

The reply is field template 1.

Key in

sequence number:	10	tab
name:	CARD NO.	tab
length:	3	tab
type:	N	

Transmit.

The reply is field template 2.

Key in

output position	1	tab, tab, tab, tab, tab
justify out:	R	

Transmit.

The reply is field template 1.

Key in

sequence number:	20	tab
name:	CARD TYPE	tab
length:	1	tab, tab, tab
end line:	E	
constant/range/value:	V	

Transmit.

The reply is field template 2.

Key in

output position:	4	

Transmit.

The reply is value constant template.

Key in

I

D

A

Transmit.

The reply is field template 1.

Key in

sequence number:	30	tab
name:	CONTENTS	tab
length:	20	

Transmit.

The reply is field template 2.

Key in

output position:	5	tab
write after:	C	

Transmit.

The reply is field template 1.

Press FORMS key (to cancel forms mode)

Key in

/PRI WORK AREA

The reply is

WORK AREA PRINTED

Key in

/DIS

The reply is the format layout as the Edit would display it.

Key in

/SAV

The reply is

G1 FMT1 SAVED

Key in

/PD G1

The reply is the directory for group G1, showing the single format FMT1.

Key in

/DEL

The reply is

GROUP AND FORMAT DELETED

Key in

/GET G1 FMT1

The reply is

G1 FMT1 NOW AVAILABLE

Key in

/DUM DUMP/FILE G1

The reply is

G1 DUMPED

Key in

/LIS

The reply is group template 1, with data for G1.

Transmit to receive the next page, and continue pressing "transmit" until the whole group and format has been listed.

Key in

/AME FOR

The reply is format template, with the data for FMT1.

Key in tab

next format: FMT1

Transmit.

The reply is

FORMAT AMENDED

Key in

/TER

The reply is

END OF SESSION

4.10 Worked Example 2

The group MULT contains four formats, which show how to take full advantage of the Write After and Print After options. Format MULTIOUT shows how a format can be set up to produce multiple output records from one input record. Formats MULTIIN1 and MULTIIN2 show how one output record can be produced from multiple input records. Format MULTIALT shows how the first part or second part (or both parts) of an input record can produce an output record (or records). The first and fourth formats demonstrate facilities which are particularly useful for the TD802 terminal, so that the 1920 characters of the screen can be fully utilised. The other formats demonstrate a facility which may be useful for the TD701 terminal, where the 256 characters of the screen may be insufficient for the data plus field names plus control information associated with one output record. Details of these formats may be found on pages 4-46 to 4-49.

The MULTIOUT format produces up to three output records from each input record. Card type, vendor number and name are duplicated (their values are placed in the duplication buffer), but part and price are different in each record. Fields 40 and 50 will represent part and price in the first record, fields 60 and 70 will represent part and price in the second record, and fields 80 and 90 will represent part and price in the third record. After the third record is written the duplication buffer is cleared. The same applies to the journal. It should be noted that fields 60 - 90 are not mandatory, and if the data input for those fields is blank, then only one output record is written, and only one line is printed in the journal.

Formats MULTIIN1 and MULTIIN2 produce the same information as MULTIOUT. Card type, vendor number and name are input using MULTIIN1, but no output record is written at this stage. Format MULTIIN2 automatically follows MULTIIN1, and contains part and price. The input of data for price causes an output record to be written, but the duplication buffer is not cleared. MULTIIN2 automatically follows itself, and fresh data for part and price replace the old data, while not changing card type, vendor number and name.

Format MULTIALT provides the ability to produce an output record containing a customer name and telephone number, or a record containing customer address, or both. Field 30 sets the Write After option to Save, so that a record will not be written after field 50 unless data is keyed in for name and telephone number. Field 60 overwrites the record identifier 1 with a 2 (to indicate that what follows is an address rather than a name), and once more sets the Write After option to Save. If the data for fields 70 - 90 is blank, then the record with the customer address will not be written.

These formats are not intended to be studied as examples of the best way to construct data entry formats, but they should demonstrate a few of the things that can be done, and most users should find that a format based on MULTIOUT provides a very efficient means of data entry.

B1700 DATA ENTRY SYSTEM F O R M A T M A I N T E N A N C E R E P O R T TIME 10:02 DATE 12/03/77 PAGE 4

FORMAT NAME MULTIOUT NEXT FORMAT MULTIOUT SECURITY 55 D-VERIFY DATE LAST CHANGED 15/09/76

SEQ	NAME	LGTH	TYP	DUP IN	END LINE	CNS/VAL /RANGE	XFT	CDV	BTOT	MAND	JUST IN	OUT PCS	BRT AFT	DUP OUT	BLNK FILL	PACK	JUST OUT	SIGN	SIGN CONV	PRINT COL	PRINT AFTER	PRINT NAME
10	CARD TYPE	3				C						1		D						1		
	CONSTANTS					VEN																
20	VEND.NO.	6	N	D	E			1		M		4		D		P				5		P
30	NAME	20		D	E							7		D						20		
40	PART.	6								M		27								43		P
50	PRICE	10	N		E				2	M		33	W			P	R			57	P	P
60	.	6										27								43		P
70	.	10	N		E				2			33	W			P	R			57	P	P
80	.	6										27								43		P
90	.	10	N		E				2			33	C			P	R			57	C	P

**** TOTALS **** FIELDS INPUT 9 FIELDS OUTPUT 9 FIELDS PRINTED 9
INPUT DATA SIZE 77 CHARACTERS OF CONSTANT DATA 3

Figure 4.1 Format MULTIOUT

```
B1700 DATA ENTRY SYSTEM      F O R M A T   M A I N T E N A N C E   R E P O R T      TIME 10:03  DATE 12/03/77  PAGE   2
----------------------      -----------------------------------------------      ----          ----                ----

   FORMAT NAME MULTIIN1          NEXT FORMAT MULTIIN2   SECURITY  50     O-VERIFY            DATE LAST CHANGED 15/09/76

SEQ    NAME    LGTH TYP CLP END CAS/VAL XFT CDV BTGT MAND JUST    CRT NRT DUP BLNK PACK JUST SIGN SIGN   PRINT PRINT PRINT
                    IN  LINE /RANGE                      IN     POS AFT OUT FILL      CLT      CONV   COL   AFTER NAME
 10 CARD TYPE    3          E     C                              1       D                              1
                CONSTANTS   VEN
 20 VEND.NO.     6  N       E              1        M             4       D        P                     5          #
 30 NAME        20                                               7       D                             20

**** WARNING ****  THE BUFFER IS NOT WRITTEN AFTER LAST FIELD
**** TOTALS  ****            FIELDS INPUT    3              FIELDS OUTPUT   3            FIELDS PRINTED    3
                             INPUT DATA SIZE   29           CHARACTERS OF CONSTANT DATA    3
```

Figure 4.2 Format MULTIIN1

B1700 DATA ENTRY SYSTEM F O R M A T M A I N T E N A N C E R E P O R T TIME 10:03 DATE 12/03/77 PAGE 3

FORMAT NAME MULTIIN2 NEXT FORMAT MULTIIN2 SECURITY 50 O-VERIFY DATE LAST CHANGED 15/09/76

SEQ	NAME	LGTH	TYP	DUP IN	END LINE	CNS/VAL /RANGE	XFT	CDV	BTCT	MAND	JUST IN	COL POS	WRT AFT	DUP OUT	BLNK FILL	PACK	JUST CLT	SIGN	SIGN CONV	PRINT COL	PRINT AFTER	PRINT NAME
40	PART.	6			E					M		27								43		P
50	PRICE	10	N						2	M		33	N			P	R			57	P	P

**** TOTALS **** FIELDS INPUT 2 FIELDS OUTPUT 2 FIELDS PRINTED 2
INPUT DATA SIZE 16 CHARACTERS OF CONSTANT DATA 0

Figure 4.3 Format MULTIIN2

```
B1700 DATA ENTRY SYSTEM      F O R M A T   M A I N T E N A N C E   R E P O R T      TIME 10:03  DATE 12/03/77  PAGE   5

   FORMAT NAME MULTIALT          NEXT FORMAT MULTIALT   SECURITY  60     O-VERIFY          DATE LAST CHANGED 15/09/76

SEQ    NAME   LGTH TYP CUP END CNS/VAL XFT CDV BTGT MAND JUST  CRT  WFT DUP BLNK PACK JUST SIGN SIGN   PRINT PRINT PRINT
                    IN  LINE /RANGE                      IN   POS  AFT OUT FILL      CLT      CONV   COL   AFTER NAME
 10              3                 C                          1        D                              1
                  CONSTANTS   CUS
 20              1                 C                          4                                       5
                  CONSTANTS   1
 30              6  N           E                 2        M   5    S   D                              10    S
 40 NAME.       24              E                             11                    L                  20          B
 50 TEL..       15              E                             15    N                                  60    P     P
 60 .CUS..       1              E      C                      4    S                                   5    S
                  CONSTANTS   2
 70 ADDR.       24              E                             11                    L                  20          P
 80     .       24              E                             15                    L                  55
 90     .       20                                            19    N               L                  85    C

**** TOTALS ****          FIELDS INPUT    9          FIELDS OUTPUT   9          FIELDS PRINTED    9
                          INPUT DATA SIZE  118       CHARACTERS OF CONSTANT DATA    5
```

Figure 4.4 Format MULTIALT

5. THE EDIT PROGRAM

5.1 GENERAL

The part of the system that is used most in normal operation is the Edit program. This perforns the actual entry and checking of data. For this reason the data entry commands have been formulated so as to be easy to understand and easy to use. All the commands are explained in detail at the end of this chapter. The first sections of the chapter deal with the basic batch procedures.

In the explanation of Edit commands a distinction may be made between the data entry commands /ACCEPT - // (VERIFY) and the control commands /BREAK - /WRU. All the commands in the latter group may be entered either with or without an operator being logged in. For the data entry commands, there are obviously situations where certain commands may or may not be used. For instance, a batch must be open before a format can be displayed and an operator must be logged in before a batch can be opened. Whilst it is self-evident at which specific point some commands - /HELLO or /OPEN for example - may be used. The following table may help to clarify the restrictions as regards usage of data entry commands.

Restrictions by command:	Operator logged in	Batch open, or in inquiry mode	Modes allowed (EVDI)
Accept (/ACCEPT)	YES	YES	EV
Delete (/DELETE)	YES	YES	EV
Display (/<record number>)	YES	YES	EVD
Enter (//<format>)	YES	YES	E
Inquiry (//<format>)	YES	YES	I
Insert (/ INSERT)	YES	YES	EV
Modify (//<record number>)	YES	YES	E
Verify (//<record number>)	YES	YES	V
Hello (/HELLO)	NO	NO	
Bye (/BYE)	YES	NO	
Release (/RELEASE)	YES	no restrictions	
Open (/OPEN)	YES	NO	
Close (/CLOSE)	YES	YES	all modes
Status (/STATUS)	no restrictions		
All control commands	no restrictions		

E = Enter mode, V = Verify mode,
D = Display mode, I = Inquiry mode

The various elements of each command are either key words, for instance VERIFY, or modifiers, which are enclosed in angular brackets, for instance <operator number>. The underlined part of a key word is sufficient to identify it.

In the command:

/HELLO <operator number><password>

it is only necessary to enter /HEL followed by a valid operator number and a valid password.

Where square brackets appear, the information within the brackets is optional:

[<format name>]

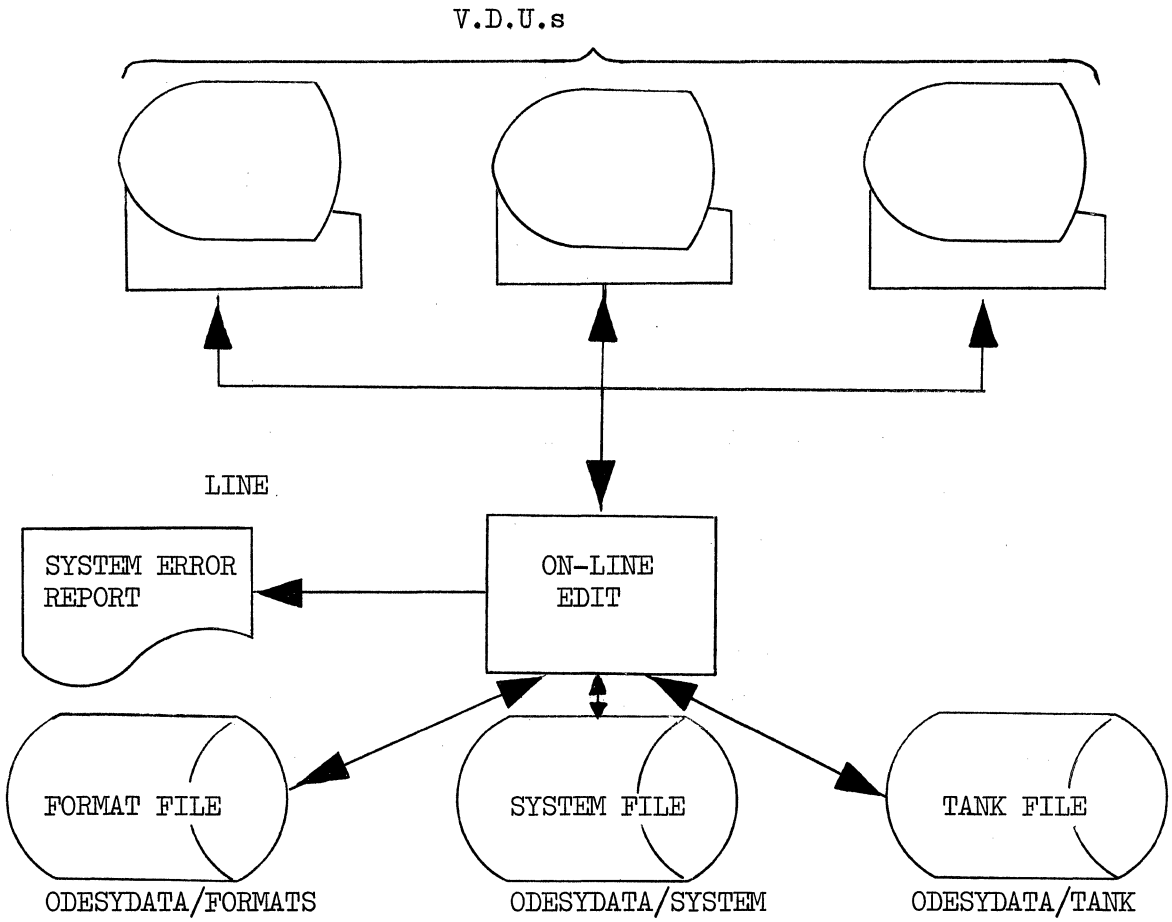
Where braces appear, one of the expressions within the braces must be selected:

/OPEN { [<batch number>] ENTER [<group>] | <batch number> [VERIFY] | INQUIRY <group> }

The above example shows a combination of braces and square brackets. In this case, one of the 3 expressions within the braces must be selected, but having selected, say, the second expression, then the keyword VERIFY is optional.

There are certain commands that may be restricted to supervisors only. These are /ACCEPT, /DELETE, /INSERT and /RELEASE of the data entry commands and /CONTROL and /TERMINATE of the control commands.

5.2 EDIT PROGRAM BLOCK DIAGRAM



5.3 DATA ENTRY

Each document or a group of documents in a batch has a corresponding data entry format. The operator requests the appropriate format by entering

// <format name >

The Edit checks that the specified format exists and builds the screen format from the specifications on the Formats file. The format is displayed at the station and the operator fills in the unprotected data areas. Many fields may have default values associated with them which will be displayed as part of the screen format. The operator may tab over them if they are correct or may enter the data if those values are not appropriate. When the format has been filled in the operator transmits it to the Edit. The Edit performs the requested audit checks, accumulates batch totals and crossfooting and, if the data is correct, adds the data to the batch. If there are errors in the data the Edit marks each field or character in Error with a question mark and sends the format back to the operator. There is a "number of errors" indicator on the first line of the screen (see figure 5.1) which shows the number of errors detected. If a crossfoot total error was found the number of errors is followed by an "X" (i.e. "ERR=02X"). The operator has two options. He or she may be capable of correcting the error immediately, in which case the question marks are replaced by the correct characters, or the operator may not be able to correct the errors, in which case the operator may force the record to be accepted using /ACCEPT and make a note

for that record to be corrected during verification. If an operator forces a record to be accepted batch totals are accumulated for all appropriate fields regardless of whether or not those fields were in error.

If a document has been omitted during data entry it may be inserted in its proper place using the INSERT command (see 5-28). Conversely, if a document has been included erroneously then it may be deleted from the batch by using the DELETE command (see 5-22). If a document has been entered incorrectly, then it may be corrected by using either the MODIFY command (see 5-29) or the VERIFY command (see 5-39).

```
123  0012 PAYRMNTH  ENT                    ERR = 1

EMP NAME[ JOHN A. ANON                        ]
EMP NUMB[ 123456              ]
HRS WORKED[ 375             ] OVR TIME[       ]
RATE[ ?      ] TAX CODE[ 03         ]




                                              [ ]
```

Figure 5.1 - Example of Error Display

5.4 OPENING A BATCH

Before data in a batch can be entered, verified or displayed an operator must open the batch. This connects the operator and the batch together preventing more than one operator from accessing the data in that batch. It also adjusts any batch totals which are being accumulated and sets certain pointers in the station table. Only one batch may be open at a station at one time.

ENTER: Typically a batch of documents is given a batch number by the data control department and the batch is given to an operator to be entered. The operator logs in at a station and keys in /OPEN <batch number> ENTER <group> where <group> corresponds to the application being entered, e.g. payroll, inventory or accounts receivable. If a batch of the specified number is already allocated then the Edit will open the old batch and any data entered will be added to the end of the batch. If the specified batch number does not exist then the new batch is allocated. A <group> is not required in the case where previously allocated batch is being added to; the formats used must be in the same group as was specified when the batch was first allocated. Entry may then proceed (see 5.3)

VERIFY: To open a batch for verification the operator enters

/OPEN <batch number> VERIFY

The group and formats have already been defined during data entry. The operator may then proceed to verify the batch (see 5.5).

DISPLAY: To open a batch for display the operator enters

/OPEN <batch number>

The group and formats have already been defined during data entry. The operator may then proceed to display selected records.

INQUIRY: To open in inquiry mode the operator enters

/OPEN INQUIRY <group>

No batch is allocated in inquiry mode. Inquiry mode allows an operator to directly access a data base via a user program.

5.5 DATA VERIFICATION

In order to ensure that the data in a batch is correct a second operator may be required to re-key each record in the batch. The Edit will compare what was entered previously with this data and report any comparison errors. The syntax for verify is:-

// <record number>

There are two options which the user may select which are relevant to verification. First, during format creation the user may specify that certain formats do not require verification. This means that the Edit will not display a record created under such a format. This applies both to an explicit request for a record and to automatic sequence mode (implicit request). Second, during format creation the user may also specify whether or not the data previously entered is to be displayed on the screen for verification. This would allow an operator to visually verify the data and transmit it back to the Edit.

There are two situations where errors can occur in the VERIFY mode. Since two sets of data are being compared for verification, either the data keyed in during ENTER mode or the data keyed in during VERIFY mode might contain errors.

If the ENTER data is incorrect and the correct VERIFY data is keyed in, the display will show a comparison error. The operator must key in the correct VERIFY data again and the record will be accepted.

If the VERIFY data is incorrect the operator must key in the correct data twice more. This is because the incorrect VERIFY data will be retained for comparison with whatever is keyed in next, therefore there will be comparison errors the first time, even though the data is correct.

The two methods may be illustrated as follows:-

ENTER data incorrect

Step	Action
1.	Key in ENTER data incorrectly.
2.	Close batch. Open batch in VERIFY mode.
3.	Request record for display using //<record number>
4.	Note that ENTER data is incorrect, key in correct VERIFY data
5.	Comparison error displayed
6.	Key in correct VERIFY data again
7.	Record accepted

2. VERIFY data incorrect

Step	Action
1.	Key in correct ENTER data - transmit
2.	Close batch. Open batch in VERIFY mode
3.	Request record for display using // <record number >
4.	Key in incorrect VERIFY data - transmit
5.	Comparison error displayed
6.	Key in correct VERIFY data - transmit
7.	Comparison error displayed
8.	Key in correct VERIFY data - transmit
9.	Record accepted

5.6 RELEASING A BATCH

A Supervisor indicates to the ODESY system that a batch is ready to be extracted to an Application file by releasing it. This schedules it to be extracted by the EXTRACT program at its next execution.

There are two basic forms of the release command

- the standard release command and the extended release command.

The format of the standard release command is:

/RELEASE [DELETE / OVERRIDE]

In order to release a batch using the standard release command the supervisor must first open it. It is intended that the first record will be inspected to ensure that the batch is the correct one. The supervisor then enters /RELEASE. The Edit will either respond with a message indicating that the batch is now scheduled for release or a message indicating that there are batch total comparison errors. If there are errors the Edit will not schedule the batch for extraction unless the errors are corrected or the supervisor enters /RELEASE OVERRIDE. The supervisor must then close the batch.

If a batch has been fully processed by ODESY and the Application System, it is necessary to delete the batch from the Tank file to make the record space available for other batches. Again, the supervisor must open the batch to ensure that this is the correct one, then enter /RELEASE DELETE and close

the batch. The batch is now scheduled for deletion by the EXTRACT program and the records will be physically deleted at its next execution. A batch may be retrieved from its scheduled extraction or deletion at any time up to the running of the EXTRACT program simply by opening it again.

The extended RELEASE command provides a supervisor with a method of releasing a single batch or a series of batches without having to open and close each batch separately. This makes releasing a batch a much easier operation, however it provides more scope for error because the supervisor is not able to inspect any of the data in the batch before releasing it. The user may exclude the extended RELEASE command if necessary (See section 3.4 and section 2.2). The format of the extended RELEASE command is:

/RELEASE <batch number> [<batch number 2>] [DELETE / OVERRIDE]

In order to release a single batch, the supervisor enters /RELEASE <batch number>. The Edit replies with a message indicating that the batch is scheduled for release, or that there are batch total comparison errors, or that the batch is open at another station. The OVERRIDE and DELETE parameters may be used in the same way as for the standard RELEASE command.

In order to release a series of batches, the supervisor enters /RELEASE <batch number> <batch number 2>. The Edit then attempts to release all batches in the range <batch number> to <batch number 2>inclusive, and gives a list of batches scheduled. If there were batches with

batch total comparison errors or if they were in use at another station, then some of the batches may not be scheduled for release. The OVERRIDE and DELETE parameters may be used in the same way as for the standard RELEASE command.

5.7 BATCH CORRECTION AFTER RELEASE

It may happen that not all input verification can be done using the Data Entry System and that a separate user written input control program may be necessary to do extended checking. If this program should find errors in the input data ODESY may be used to correct these errors.
The input control program would indicate which records contained errors and an operator would re-open the batch in verify mode, request the records in error and correct the data. The batch would then again be extracted and checked by the input control program. When the batch had finally been successfully processed it would be deleted from the Tank file (see section on Release).

5.8 BATCH TOTALS

In order to use batch totals, it is necessary to create a batch header format, and one or more formats for the detail records (which will contribute to the batch totals). In a batch header format (format type = B), those fields with the batch total option set are the operator-entered batch totals. In a format for a detail record, those fields with the batch total option set contribute to the computer-calculated batch totals. When the batch is released, the operator-entered totals are compared with the computer-calculated totals.

The batch header record must be the first record in the batch, and it must be entered using a batch header format. If record 1 is not a batch header, it is assumed that batch totals are not to be checked for this batch, and if a batch header format is used to enter a record other than record 1, it will be ignored for the purposes of batch totals.

If the operator receives message 39, BATCH TOTAL COMPARISON ERROR, in reply to a release command, the batch total error should be corrected before the operator attempts to release the batch again. The computer-calculated totals may be inspected by entering /STATUS TOTALS, and the operator-entered totals may be inspected by displaying record 1 of the batch.

5.9 USE OF INQUIRY MODE

Section 10 describes the inquiry facilities available in ODESY.

In all cases, the data base or master file is accessed by a program written by the user. This user program supplies data or checks data under the control of the formats requested by the operator. In many cases, the inquiry facilities are used to assist the operator in entering a batch. Some data may be supplied directly from the data base rather than being entered by the operator, and the data entered may be checked for validity against the data base. However the inquiry faciltities may also be used for on-line updating of a data base, or for pure inquiry, where a batch is often unnecessary. In such cases, inquiry mode may be used.

To get into inquiry mode, the operator keys in an open command:

/OPEN INQUIRY <group>

The operator may now use the inquiry command to request any format in the specified group:

//<format>

Detailed operating procedures must be defined by the user, because they depend upon the design of the user program and the formats.

5.10 EXPLANATION OF INDIVIDUAL COMMANDS

The following section contains a detailed explanation of both data entry and control commands. It explains the correct syntax for the command, the use of the command, the reply if there are no errors, the error messages which might occur and an explanation of what might cause error conditions.

ACCEPT

Syntax: /ACCEPT

Use: to force the system to add a record to a batch despite any errors it may contain. It may be used when entering or inserting records. Batch totals are accumulated on the appropriate fields whether or not those fields were in error.

Reply:	Record accepted	: msg 6
Error msgs:	Not in enter or verify mode	: msg 10
	No record current	: msg 23
	No batch open	: msg 18

Error conditions:

msg no.	Text	Cause
23	THE RECORD DOES NOT EXIST	There was nothing wrong with the last record entered
18	NO BATCH IS OPEN	The operator has not "OPEN"ed a batch.
10	YOU ARE NOT ENTERING DATA	Batch was not "OPEN"ed ENTER or VERIFY

BYE

Syntax: /BYE [< operator number >]

Use : to terminate an operator's use of the system. Without the optional operator number the operator at the station entering the request is logged out. With the optional operator number the specified operator is logged out (if logged in) no matter which station he/she is using, and any open batch is closed. This would be used in the case where a station physically fails.

		msg. no.
Reply: without opr no.	You are logged out	25
with opr no.	Done	53
Error msgs:	No operator logged in	12
with opr no. only	Batch still open	17
with opr no. only	Expecting a number	54
with opr no. only	Invalid operator	14

Error conditions:

msg no.	text	cause
12	NO OPERATOR IS LOGGED IN	The operator to be logged out was not logged in.
17	THAT BATCH IS IN USE ELSEWHERE	The batch must be closed before the operator can log out.
54	I WAS LOOKING FOR A NUMBER	Operator number specified was not numeric.
14	YOUR NAME ISN'T ON FILE	Operator number specified is not a valid operator.

CLOSE

Syntax /CLOSE

Use: to close a batch when the operator has finished working with it, or to get out of inquiry mode.

Replies:	batch closed	msg no. 5
	done (end of inquiry mode)	msg no. 53
Error msg:	no batch open	msg no. 18

Warning (in verify mode only)	not all recs verified	57
and	first unverified record	69

Error conditions

msg no.	text	cause
18	NO BATCH IS OPEN	The operator has not "OPEN"ed a batch.
57	BATCH CLOSED - NOT ALL VERIFIED	Given in Verify mode when some of the records in the batch are not yet verified. This is a warning only. The batch is closed normally.
69	FIRST UNVERIFIED RECORD	Sent with message 57.

DELETE

Syntax: /DELETE <record number >

Use: to remove a record from a batch (enter mode or verify mode only)

		msg nr
Reply:	record deleted	21
Error msgs:	expecting a number	54
	record not found	23
	no batch open	18
	not enter mode or verify mode	56

Error conditions:

msg no.	text	cause
18	NO BATCH IS OPEN	The operator has not "OPEN"ed a batch.
23	THE RECORD DOES NOT EXIST	The specified record was not found in the batch.
54	I WAS LOOKING FOR A NUMBER	no <record number> was specified or it was not numeric.
56	YOU ARE IN DISPLAY MODE	The batch was not opened specifying ENTER mode or VERIFY mode.

DISPLAY

Syntax:

	Case 1 (a single blank character)
/ { ± } [< increment >]	Case 2
/ <record number>	Case 3

Use: to display selected records in a batch.

Case 1 - displays the next record in the batch or repeats the last increment. Used for "paging" through a batch.

Case 2 - allows "paging" through a batch using the given increment or 1 if no increment is specified.

Case 3 - the record number specified is displayed (if it exists)

Reply: display of record

Error msgs:	msg no.
no open batch	18
record doesn't exist	23
invalid syntax (not ±)	35
expecting a number (increment not number)	54
system error (missing format)	04

Error conditions:

msg no.	text	cause
18	NO BATCH IS OPEN	The operator has not "OPEN"ed a batch.
23	THE RECORD DOES NOT EXIST	The specified record number doesn't exist in the batch or adding the increment to

		the current record number gives a record number outside the range of the batch, or there is no current record.
35	INVALID SYNTAX	"+" or "-" not entered correctly.
54	I WAS LOOKING FOR A NUMBER	<record number >or <increment >was not numeric.
04	SYSTEM ERROR - LAST COMMAND LOST	The format specified for displaying the requested record couldn't be found on the Formats file. It is not possible to display or verify that record.

ENTER

Syntax: // <format>

Use: // <format> is used for the entry of data into a batch.

It is keyed in immediately after /OPEN [<batch no.>] ENTER.

Reply: display of requested format.

Error conditions:

Msg no.	text	cause
10	YOU ARE NOT ENTERING DATA	The batch is not open in ENTER mode.
16	THAT FORMAT DOES NOT EXIST	The format requested is not in the Formats file.
18	NO BATCH IS OPEN	The /OPEN command has not been used for the batch.
76	INQUIRY PROGRAM TABLE FULL	The format is an inquiry format, but there is no room for a new user program in the program table.

HELLO

Syntax: /HELLO <operator number > <password >

Use: to log an operator into the system. Batches may not be accessed unless an operator is logged in.

		msg no.
Reply:	you are logged in	24
Error msgs:	system closing down	33
	an operator is logged in	11
	expecting a number (opr no)	54
	invalid operator	14
	invalid syntax (missing or wrong password)	35

For message 14, a warning will be printed in the log.

Error conditions:

msg no.	text	cause
11	AN OPERATOR IS LOGGED IN	Some other operator is using this station. Enter /BYE to log that operator out.
14	YOUR NAME ISN'T ON FILE	The operator number specified is not a valid operator.
33	THE SYSTEM IS CLOSING DOWN	A supervisor has begun the system shut down procedure and no one is allowed to begin new work.
35	INVALID SYNTAX	Missing password or numeric password.
54	I WAS LOOKING FOR A NUMBER	The required operator number is missing or not numeric.

INQUIRY

Syntax : // <format>

Use: // <format> is used in inquiry mode to retrieve information from a data base, or to directly update information in a data base. It is keyed in after /OPEN INQUIRY <group>.

Reply: display of requested format. If it is a supply data format, or a check-and-supply format, the data supplied by the user program (from the data base) will also be displayed.

Error Conditions:

Msg. No.	text	cause
10	YOU ARE NOT ENTERING DATA	You are not in INQUIRY mode.
16	THAT FORMAT DOES NOT EXIST	The format requested is not in the Formats file.
18	NO BATCH IS OPEN	The /OPEN command has not been used to get into INQUIRY mode.
76	INQUIRY PROGRAM TABLE FULL	The format is an inquiry format, but there is no room for a new user program in the program table.

INSERT

Syntax: /INSERT <record number> <format>

Use: to insert a previously omitted record into a batch (enter mode or verify mode only). Insertion is made after specified <record number> Specify zero to insert record number one.

Reply: display of requested format

		msg. no.
Error msgs:	no open batch	18
	not enter mode or verify mode	56
	record number missing	54
	record not in batch	23
	format missing or non-numeric	35

Error conditions:

msg no.	text	cause
56	YOU ARE IN DISPLAY MODE	The batch was not opened specifying ENTER mode or VERIFY mode.
18	NO BATCH IS OPEN	The operator is not currently working on a batch.
23	THE RECORD DOES NOT EXIST	The record number after which the insertion was to be made is not in the batch.
35	INVALID SYNTAX	The format name to be used was missing or numeric.
54	I WAS LOOKING FOR A NUMBER	The expected record number was missing or non-numeric.
76	INQUIRY PROGRAM TABLE FULL	The format is an inquiry format, but there is no room for a new user program in the program table.

MODIFY

Syntax: //<record number>

Use: to make corrections to an existing record when in enter mode.

note: The syntax of the MODIFY command is exactly the same as the syntax of the verify command, but the former may only be used in enter mode, whereas the latter is only used in verify mode.

Reply: display of the requested record.

		msg no.
Error msgs:	system error	4
	no open batch	18
	record not in batch	23
	not enter mode or verify mode	56

Error conditions:

msg no.	text	cause
4	SYSTEM ERROR-LAST COMMAND LOST	The requested format is not in the Formats file.
18	NO BATCH IS OPEN	The operator has not opened a batch.
23	THE RECORD DOES NOT EXIST	The specified record number does not exist in the batch.
56	YOU ARE IN DISPLAY MODE	The batch was not opened specifying ENTER mode.
76	INQUIRY PROGRAM TABLE FULL	The format is an inquiry format, but there is no room for a new user program in the program table.

OPEN

Syntax:

Case 1	/OPEN	<batch number>	[VERIFY]	
Case 2	/OPEN	[<batch number>]	ENTER	[<group>]
Case 3	/OPEN	INQUIRY	<group>	

Use: to make a batch available at a station for inspection or modification, or to get into inquiry mode.

Case 1 - opens an existing batch for verification if VERIFY specified, otherwise for display only.

Case 2 - opens either an existing batch for additional input and insertion or a new batch to begin entry. If a <batch number> is not specified one will be supplied by the Edit. If <batch number> already exists the group need not be specified.

Case 3 - to get into inquiry mode, so that a data base may be accessed (via a user program). No batch is opened.

		msg. no.
Reply:	batch opened (cases 1 and 2)	29
	you are in inquiry mode (case 3)	70
Error msgs:	batch already open	9
	invalid syntax (nothing after open, group name required, bad mode)	35
	no free batch	37
	batch in use elsewhere	17
	system error (missing group for existing batch)	4
	group doesn't exist	36
	no such batch (verify or display)	52
	attempt to change group for an existing batch	64

Error conditions:

msg no.	text	cause
9	A BATCH IS ALREADY OPEN	The operator was previously working on a batch and has not entered /CLOSE yet.
35	INVALID SYNTAX	For case 1: the required batch number was missing Case 2: the group name was required to open a new batch. Either case: the mode specifier was mis-spelt or numeric.
17	THAT BATCH IS IN USE ELSEWHERE	The batch that was requested in the OPEN is being used at another station. /STATUS <batch number> will indicate which.
37	THERE ARE NO FREE BATCHES	The Edit has been requested to allocate a batch and it cannot find one that is free. The only way to make space available is to /RELEASE DELETE a batch and then run the Extract.
36	THAT GROUP DOES NOT EXIST	The group name specified cannot be found on file.
52	THE BATCH DOES NOT EXIST	The batch requested for Verify or Display mode has not been previously allocated. There are no records in the batch.
4	SYSTEM ERROR - LAST COMMAND LOST	The group name for an existing batch could not be found on the Formats file. It has been erroneously deleted.
64	YOU CANNOT CHANGE GROUP	A group name was specified for an existing batch which was different to the original group name.

RELEASE

Syntax:

/RELEASE [DELETE / OVERRIDE] Case 1

/RELEASE <batch number> [DELETE / OVERRIDE] Case 2

/RELEASE <batch number ><batch number 2 > [DELETE / OVERRIDE] Case 3

Use: to schedule a batch (or batches) for extraction or to schedule a batch (or batches) for deletion from the Tank file. When DELETE and OVERRIDE are omitted, batch total comparisons are done, and the batch will not be scheduled if errors are found. By specifying OVERRIDE, batch total comparison errors are ignored. When DELETE is specified, the batch is marked "to be deleted", and the Extraction Program physically deletes the records at its next execution.

Case 1 is the standard RELEASE command. In this case there must be a batch open. Releasing the batch does not close it. The operator must still enter /CLOSE before beginning new work.

Cases 2 and 3 comprise the extended RELEASE command. In each case it is not necessary to have a batch open because the batch or batches to be released are specified in the command. In case 2 a single batch is scheduled for extraction or deletion. The batch will not be scheduled if it is open at another station.

In case 3 a series of batches, from <batch number> to <batch number 2> inclusive, is scheduled for extraction or deletion. In this case a list of batches scheduled is sent to the operator. A batch will not be scheduled if it is in use at another station, and unless OVERRIDE is specified, it will not be scheduled for extraction if there are batch total comparison errors.

		msg. no.
Replies:	batch released	7
	batch deleted	19

Case 3 only:

The list of batches scheduled consists of a table of entries of the following form:

xxx yz

where xxx is the batch number, y is the release result and z is the delete result. There will be an entry in the table for each allocated batch in the range <batch number> to <batch number 2> inclusive.
y and z take the following values:

B - batch total comparison error
D - scheduled for deletion
I - in use at another station
R - batch has been released
S - scheduled for extraction
blank - not scheduled

If batches are being scheduled for extraction, y will be S, B or I, and z will be blank or D. y indicates whether the batch was scheduled, and z indicates the current delete status.

If batches are being scheduled for deletion, z will be D or I, and y will be blank, S or R. z indicates whether the batch was scheduled and y indicates the current extraction status.

		msg. no.
Error msgs:	batch total comparison error	39
	no open batch (case 1 only)	18
	batch in use elsewhere (case 2 only)	17
	no such batch (case 2 only)	52
	invalid syntax	35

Error conditions:

msg. no.		cause
17	THAT BATCH IS IN USE ELSEWHERE	Case 2 only. The batch specified is being used at another station.
18	NO BATCH IS OPEN	Case 1 only. The operator is not currently working on a batch.
35	INVALID SYNTAX	DELETE or OVERRIDE have been mis-spelt.
39	BATCH TOTAL COMPARISON ERROR	Cases 1 and 2. The Edit accumulated batch totals do not agree with the totals entered in the batch header. See OVERRIDE above.
52	THE BATCH DOES NOT EXIST	Case 2 only. The batch specified does not exist.

STATUS

Syntax :

/STATUS		case 1
/STATUS	<batch number>	case 2
/STATUS	TOTALS	case 3
/STATUS	ALL	case 4
/STATUS	<batch number> ALL	case 5

Use : to display the status of a station, a batch, or the batch totals of the current batch.

Case 1 - displays the status of the requesting station.

Case 2 - displays the status of the batch specified.

Case 3 - displays the status of the batch totals of the current batch.

Case 4 - allows "paging" through station status reports for all stations, in LSN order. The status of the first station is displayed in response to the STATUS command, and subsequent "pages" are requested by transmitting a single blank character.

Case 5 - allows "paging" through batch status reports for all allocated batches, beginning with the batch number specified. The status of the batch specified is displayed in response to the STATUS command, and subsequent "pages" are requested by transmitting a single blank character. If the batch specified is not allocated, the status of the first allocated batch following the one specified will be displayed. Specify one to obtain status of first allocated batch in the system.

STATUS (Cont'd)

Replies :

	msg. no.	
batch status		
batch no. + no. recs	40	
inuse at station	41	
enter status	42	(1)
verify status	43	(2)
delete status	44	(3)
extraction status	45	(4)
opened and released by	46	

(1) possible values : IN PROGR or COMPLETED
(2) possible values : blank, COMPLETE and IN PROGR
(3) possible values : blank, DELETED
(4) possible values : blank, SCHEDULED and DONE

station status	
sta no. + operator	47
current + last batch	48
current rec no. + mode	49
current group + format	50
log in time	51

STATUS Cont'd

batch totals status (for current batch)

1 <signed value>
2 <signed value>
3 <signed value>
4 <signed value>
5 <signed value>

the signed value is always 15 digits plus a sign (zero-filled and right justified).

Error msgs :	msg no.
expecting a number	54
no such batch	52
no batch totals in inquiry mode	70

Error conditions:

msg no.	text	cause
52	THE BATCH DOES NOT EXIST	The specified batch has not been allocated. It contains no records.
54	I WAS LOOKING FOR A NUMBER	The expected batch number was not numeric. Also occurs if TOTALS was not keyed in correctly.
70	YOU ARE IN INQUIRY MODE	The operator is in inquiry mode, so there is no batch open, and there can be no batch totals.

VERIFY

Syntax: //<record no.>

Use: //<record no.> is used in order to verify the data for this record, either by performing a visual check or by rekeying the data (see OPEN).
Note: the syntax of the VERIFY command is exactly the same as the syntax of the MODIFY command, but the former may only be used in VERIFY mode and the latter may only be used in ENTER mode.

Replies: If the format was set up with no verification required, the next record requiring verification is searched for. If all records upto the end of the batch are not to be verified, the reply is:

Msg. No.	Text
53	DONE

If verification is required the format is displayed so that data may be rekeyed. For display verification the format is displayed showing the data keyed in during enter mode so that a visual check may be made.

Error conditions:

Msg No.	Text	Cause
04	SYSTEM ERROR - LAST COMMAND LOST	The requested format is not in the Formats File.
18	NO BATCH IS OPEN	The OPEN command has not been used for the batch.

Error conditions:

Msg no.	text	cause
23	THE RECORD DOES NOT EXIST	The record number specified does not exist in this batch.
56	YOU ARE IN DISPLAY MODE	The batch was not opened specifying verify mode.
76	INQUIRY PROGRAM TABLE FULL	The format is an inquiry format, but there is no room for a new user program in the program table.

BREAK

Syntax : /BREAK [<lsn>]

Use : to stop the network controller from trying repeatedly to transmit a message to a station which has refused the message. A station may enter /BREAK for itself (in which case the <lsn> need not be specified) or for another station. The Edit will make the station "NOT READY", remove the "offending" message and then ready the station so the rest of the messages queued for that station may be sent.

Reply : none

Error conditions:

msg no.	text	
54	I WAS EXPECTING A NUMBER	The expected station number was not numeric.
31	INVALID STATION NUMBER	The <lsn> quoted was not a valid station number.

BUG

Syntax : /BUG [ON / OFF]

Use : for debugging purposes only. Not available during normal program execution. This command turns off and on the debugging trace of program execution. By entering only /BUG the current status of this option will be displayed. Any modifier besides ON will turn BUG off.

Replies : BUG { ON / OFF }

Error conditions:

msg no.	text	cause
30	I DON'T RECOGNIZE THAT COMMAND	This version of the Edit does not include the debugging facility.

CONTROL

Syntax: /CONTROL <lsn>

Use: to make a station a control station enabling it to use the TERMINATE and DS commands. The central machine console is always a control station. A station can be made a permanent control station by setting the CONTROL STATUS option when specifying the station to Filemaint (see section 3.5). Otherwise a station is only a control station for the duration of the current execution of the Edit.

Any control station can make any other station a control station.

Reply:

msg. no.	text
34	MADE CONTROL STATION

Error conditions:

msg. no.	text	cause
20	THAT COMMAND IS RESTRICTED	The requesting operator is not a supervisor or that station is not a control station.
31	INVALID STATION NUMBER	<lsn> specified was not a valid station number.

DS

Syntax : /DS <program name>

Use : causes the termination of a user program. The user program will receive an end of file condition for the remote file. The station must be a control station.

Reply :

msg. no.	text
53	DONE

Error conditions :

msg. no.	text	cause
20	THAT COMMAND IS RESTRICTED	The station is not a control station
35	INVALID SYNTAX	The program name is missing, or incorrectly specified.
65	INQUIRY PROGRAM IS NOT RUNNING	The specified program is not running

DUMP

Syntax : /DUMP

Use : causes a full program dump to be taken. Processing continues normally. This command is not normally available; it is for debugging purposes only.

Reply : none

Error conditions:

msg no.	text	cause
30	I DON'T RECOGNIZE THAT COMMAND	This version of the Edit does not include the debugging facility.

READY

Syntax : /READY <lsn>

Use : to make the network controller reinstate a station that has previously been marked "not ready" in the NIF. Polling will resume for that station.

Reply:

msg no.	text
53	DONE.

Error conditions:

msg no.	text	cause
54	I WAS LOOKING FOR A NUMBER	the expected station number was not numeric.
31	INVALID STATION NUMBER	<lsn> specified was not a valid station number.

SEND TO STATION

Syntax : /SS [<lsn><text> | ALL <text>]

Use : to send any message to another station under the control of the Edit. Station number zero is the Central Machine Console. Messages will be truncated if they exceed 145 characters. If /SS ALL <text> is used, the message will be sent to all stations.

Reply:

msg no.	text
53	DONE.

Error conditions:

msg no.	text	cause
31	INVALID STATION NUMBER	<lsn> specified was not a valid station number.
54	I WAS LOOKING FOR A NUMBER	The expected station number was non-numeric.
38	THE DESTINATION IS NOT ON LINE	The station is marked as being not ready and no messages may be sent to it.

TERMINATE

Syntax: /TERMINATE [QC | NOW | <program name>]

Use: This command is used to shut the ODESY Edit down in an orderly manner. Several minutes before actual shutdown a supervisor should enter /TERMINATE at a control station. This will send a warning message to each station notifying them that they should end their work. The Edit will not allow any new operators to log in after /TERMINATE has been entered. When it is finally desired to shut the system down a supervisor enters /TERMINATE NOW or /TERMINATE QC. Specifying NOW will close any batches that are open, log out all operators and stop the Edit. Specifying QC will, in addition to this, terminate the network controller. Specifying <program name> has the same effect as QC, but in addition the <program name> will be zipped to execute after the termination of the network controller.

Replies:

1. to /TERMINATE

msg no.	text
2	CLOSING DOWN-END YOUR WORK NOW

2. to /TERMINATE [QC | NOW | <program name>]

msg no.	text
3	CLOSE FORCED - YOU ARE LOGGED OUT

TERMINATE

Error conditions:

msg no.	text	cause
20	THAT COMMAND IS RESTRICTED	The operator is not a supervisor or the station is not a control station.

WHO ARE YOU

Syntax : /WRU

Use : Entering /WRU repeats the greeting message sent to each station when the Edit begins.

Reply:(the greeting message)

```
line 1:  B1700 ON-LINE DATA ENTRY SYSTEM
     2:  COMPILED XX/XX/XX   XX:XX
     3:  VERSION 1.1
     4:  TODAY IS XXXXXXXXX   XX/XX/XX
     5:  TIME NOW IS XX:XX
```

5.11 BASIC DATA ENTRY - WORKED EXAMPLE

The example given below sets out the basic procedures involved in opening a batch, entering and verifying data, closing a batch, releasing a batch etc. It is important to note that whilst several formats can be used to enter data into a batch, all these formats must belong to the same group.

For the purpose of this worked example the group name is INVC, the 2 formats are BHDR, which is a batch header format, and REC1; the batch size is 6 records including the batch header record, the batch number is 1, the operator number is 1 and the operator password is SUP1.

After the console operator has started ODESY/EDIT key in:

/HEL 1 SUP1 - transmit

The reply is the greetings message, followed immediately by the next message, which is:

YOU ARE LOGGED IN

Key in

/OPE 1 ENT INVC - transmit

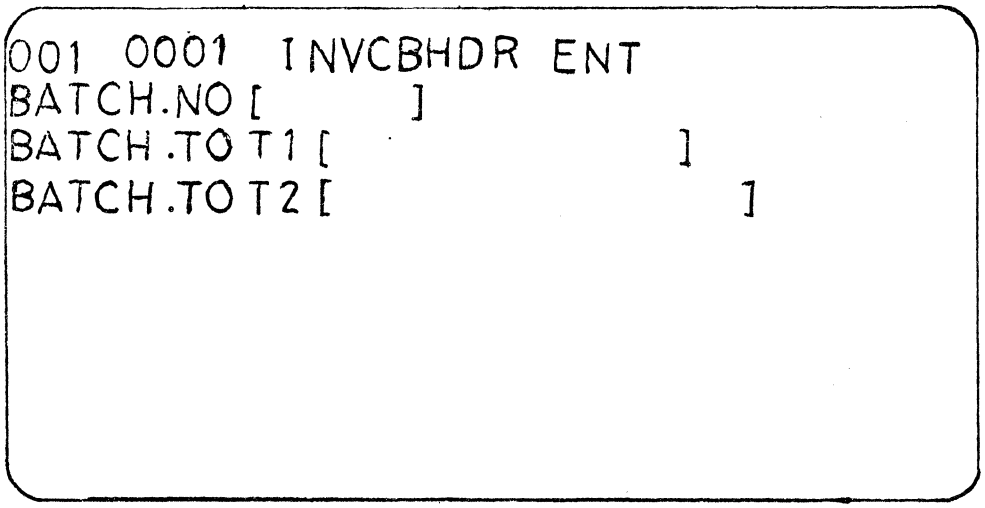
The reply is:

BATCH OPENED

Key in

//BHDR - transmit

The reply is a display of the format called BHDR which is used as a control record to accomodate a batch number and two batch totals. It appears on the screen as shown below



Key data into the fields as follows:-

BATCH.NO - 1

BATCH.TOT1 - 49

BATCH.TOT2 - 38250 - transmit

The reply is a display of a format called CREC

Key in

//REC1 - transmit

The reply is a display of REC1. Some of the fields have default values inserted into them and the user should tab to the next field if these are correct. It will also be found that both the batch number and the date are duplicated and appear automatically after they are keyed in for the first record. REC1 appears as shown below:

```
001 0002 INVCREC1 ENT
BTCH.NO2/3[      ]
STK.NO.[                    ]
ST.CODE[  ] TR.NO.[         ]
QTY.[           ] D.F[  ]
VALUE[             ]
DATE[      ] REC.CODE[  ]
AUX CODE[  ]
```

The data for the five records is as follows:-

	1	2	3	4	5
BTCH NO	1	tab	tab	tab	tab
STK NO	1007	1007	1007	1007	1007
ST CODE	tab	tab	tab	tab	tab
TR NO	5043	5062	5081	5100	5119
QTY	11	3	7	8	20
D.F.	tab	tab	tab	tab	tab
VALUE	82.50	22.50	52.50	75.00	150.00
DATE	150176	tab	tab	tab	tab
REC CODE	tab	tab	tab	tab	tab
AUX CODE	1	1	1	1	1

Key in

/CLO - transmit

The reply is

THE BATCH IS NOW CLOSED

Key in

/OPE 1 VER - transmit

The reply is

BATCH OPENED

Key in

//1 - transmit

The reply is a display of the first record, showing the data keyed in. The format has been specified with display verification. After it has been established that the data within the fields is correct, the data may be transmitted with the Cursor in its original position, that is in the left-most position of the first unprotected area of the format. If errors are detected and corrections are made the tab key should be used to bring the cursor to the parking space at the bottom right-hand corner of the screen before retransmitting the data.

After the first record has been displayed, verified and transmitted the next record appears automatically. Continue verifying and transmitting until the sixth record has been transmitted. The fact that this is the last record in the batch is signified by a reply of:

THE RECORD IS ACCEPTED

Key in

/CLO - transmit

The reply is

THE BATCH IS NOW CLOSED

Key in

/OPE 1 - transmit

The reply is:

BATCH OPENED

Now clear the screen by pressing the CLEAR HOME key and the SHIFT key simultaneously. This makes it easier to see the batch totals when they are displayed.

Key in

/STA TOT - transmit

the reply should be

1 + 000000000000049

2 + 000000000038250

3 + 000000000000000

4 + 000000000000000

5 + 000000000000000

If the totals differ from those in the batch header record, either correct the errors or use RELEASE OVERRIDE when releasing the batch (see below)

Key in

/REL - transmit

If the batch totals are correct, the reply is

NOW SCHEDULED FOR RELEASE

If not, the reply is

BATCH TOTAL COMPARISON ERROR

To force acceptance of incorrect batch totals

key in:

/REL OVE - transmit

The reply is

NOW SCHEDULED FOR RELEASE

Key in:

/STA 1 - transmit

The reply is:

BATCH NO AND RECS	001 0006
IN USE AT STATION	01
ENTRY STATUS	COMPLETED
VERIFICATION STATUS	COMPLETED
DELETE STATUS	
EXTRACTION STATUS	SCHEDULED
OPENED AND RELEASED BY	01 01

This gives a status report for the batch just entered. /STA gives a status report for the station entering the data.

6. THE EXTRACTION PROGRAM (EXTRACT)

6.1 GENERAL

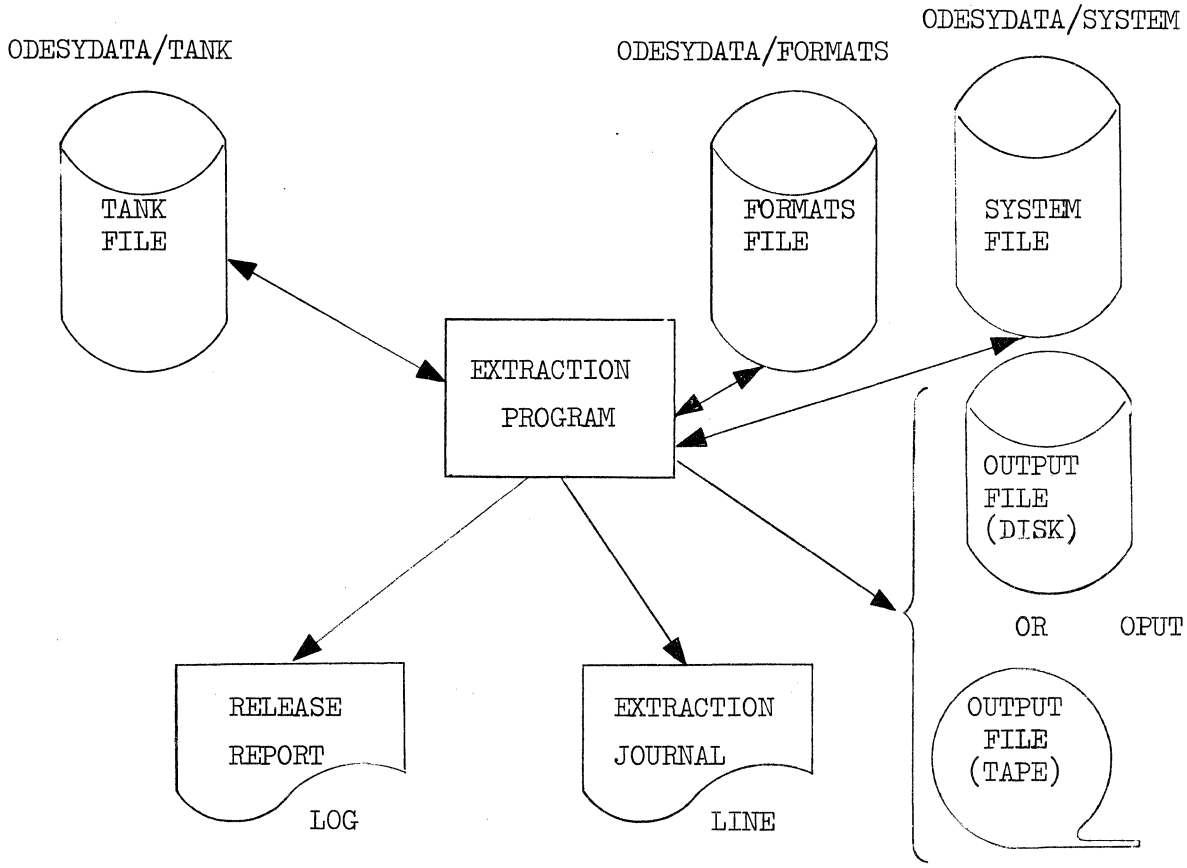
The Edit controls the entry of data, and the batches entered are stored in the Tank file. The Extraction Program takes these batches, and releases them as disk or tape files. The files released may then be used as input to Application Packages.

The Extraction Program will release a batch (or file), but will extract a record. Extraction refers to the process of reorganising the data entered into a form suitable for the Application Package, and is handled field by field and record by record, whereas release refers to the overall process of converting a batch into an output file (See Appendix D.3).

During a run of the Extraction Program, each of the 999 possible batches is scanned, and any batch that is scheduled for release or marked as deleted is examined in more detail. All such batches will be listed in the Release Report with a one line summary of the action taken

The Extraction Program may either be run at the same time as the Edit (On-line version), or separately (Stand-alone version). There is also a journal-only facility. For further information, refer to sections 6.7 and 6.8.

6.2 EXTRACTION PROGRAM BLOCK DIAGRAM



6.3 EXTRACTION AND RELEASE

When the Extraction Program finds a batch that is scheduled for release, it uses the group name to obtain details of the file to be released. If the group is not found in the Format file, a message "GROUP MISSING" is printed in the Release Report, and the batch is not released. (This can occur if the group was removed from the Format file after the batch was entered).

If it is to be a magnetic tape file, a new tape file is opened, and extracted records are written to the tape. If it is to be a disk file, then it is necessary to know if the file already exists. If it does not exist, a new disk file is created, with the appropriate record size, blocking factor, records per area and number of areas. If it does already exist, then records are added to the end of the existing file. In the latter case, a message "INCOMPATIBLE" is printed in the release report if the record size and blocking factor do not match, and the batch is not released.

If the report option is set for the group, then an Extraction Journal is produced for the batch. The content of the journal depends upon the values of the Print Column, Print After and Print Name options for each field of each format used.

Each record in the batch is entered under the control of a format, and this same format is used during the extraction of the record. If the format is not found in the Format file, a message "FORMAT MISSING" is printed in the Release Report, and the release of the batch is terminated. If it is a new disk file or a tape file, the file is purged. If it is an existing disk file, the records that have been added are removed.

The format defines what is to be done with each field in the input record. The options available are explained in the chapter on the Format Maintenance Program.

6.4 DELETION

When the Edit receives a command to delete a batch, the only action taken is to mark the batch as deleted. The actual data in the batch is not deleted until the next run of the Extraction Program. When the Extraction Program finds a batch that is marked as deleted, all data records of the batch (in the Tank file) are returned to the chain of free records, and the batch number may then be released for a new batch.

It should be noted that a batch is not automatically deleted if it is released, so that if errors are found by the Application Package, the records in error can be modified, or records inserted or deleted, and the batch can be scheduled for release again. When a batch has been satisfactorily processed by the Application Package, it should be explicitly deleted from the Tank file (using the RELEASE DELETE command).

If the user does not wish to retain released batches, then the two commands RELEASE and RELEASE DELETE should be used to force a batch to be deleted as soon as the application file has been released. A batch status report should show Extract Status = "Scheduled" and Delete Status = "Deleted" for all such batches.

If the Tank file ever becomes full, it is necessary to run the Edit to delete a few unwanted batches, and then the Extraction Program to return the data records to the chain of free records. If the Tank file regularly becomes full, it may be advisable to increase the limits of the file. See section 2.6.

6.5 STATISTICS

The Extraction Program is also concerned with gathering statistical details of the number of records entered and the number of key strokes.

It gathers by format the number of records entered for each batch released or deleted. It gathers by operator the number of records entered and verified, and the number of key strokes for enter mode and for verify mode, for each batch released or deleted. It prints (in the release report) by batch the number of records entered and verified. The data gathered by operator and by format can be printed later by the Statistics Analysis Program.

The release report also shows the number of records re-extracted, the number of records in the output file both before and after the release of the batch, and the number of lines printed (if any) in the Extraction Journal. A record is counted as being re-extracted if the batch has been released previously, and the data for the record has not been entered in the meantime.

6.6 RECOVERY

The Extraction Program has automatic recovery facilities. If there is a system failure (either hardware or software) while data is being extracted or deleted, then on running the program again, the batch in progress is properly released or deleted.

If the batch being released was to be a tape file or a new disk file, then everything is straightforward, and a new file is created. However, if the batch being released is to be added to an existing disk file, then data is added from the same point during the recovery run, provided the file still exists. The following example should illustrate what happens:

Batch 10 contains 200 records, to be added to file A, which already has 800 records. Assuming that when the system fails, only 100 records have been added to file A, then on restarting the Extraction Program, if file A still exists, the 200 records are extracted once more and they will become records 801 - 1000 of file A, even though file A had contained 900 records. However, if file A is removed from disk before restarting the Extraction Program, then a new file A is created, and batch 10 becomes records 1 - 200 of this new file.

.7 ON-LINE AND STAND-ALONE VERSIONS

There are obviously differences in operation between the two versions. In the process of releasing and deleting batches, and updating statistics, the Extraction Program updates all three files (tank, system and formats). If the Edit is running, it too will be updating the tank and system files, and the two programs must communicate with each other to synchronise their actions. If the Edit is not running, there is no problem of synchronisation.

However to the user there is no difference. The same batches are released and deleted in the same order. All that the user must decide is which version of the program is to be executed. Should the user execute the wrong version (stand-alone version while Edit is running, or on-line version when Edit is not running), an error message will be displayed at the SPO, and the program will terminate.

The same object program is used for the two versions, the only difference being the setting of an external switch. For the stand-alone version switch 1 is set to 0, whereas for the on-line version switch 1 is set to 1.

6.8 JOURNAL-ONLY FACILITY

The journal-only facility allows the user to do a trial extraction run to ensure that the correct batches are scheduled for release and deletion.

The journal-only facility is selected by setting an external switch. If switch 2 is set to 0, batches are released and deleted, and statistics are updated. However if switch 2 is set to 1, batches are not released or deleted, and no statistics are gathered. All that is produced is an extraction journal and a release report. These reports show the expected result of releasing and deleting those batches which are scheduled for release and deletion. At the top of both reports appear the words JOURNAL ONLY.

The journal-only facility is available in both the on-line and stand-alone versions of Extract:

	SW1	SW2
Stand-alone version	0	0
Stand-alone, journal-only version	0	1
On-line version	1	0
On-line, journal-only version	1	1

7. THE STATISTICS ANALYSIS PROGRAM (STATS)

7.1 GENERAL

The Statistics Analysis Program produces two reports, an operator report and a format report. Each report summarises the data entered and released since the last run of the Statistics Analysis Program, and may be used by management to identify sources of inefficiency in the system.

The formats report shows the utilisation of each format and the volume of data for each application. The number of records entered for each format is shown, and there is a total for each group.

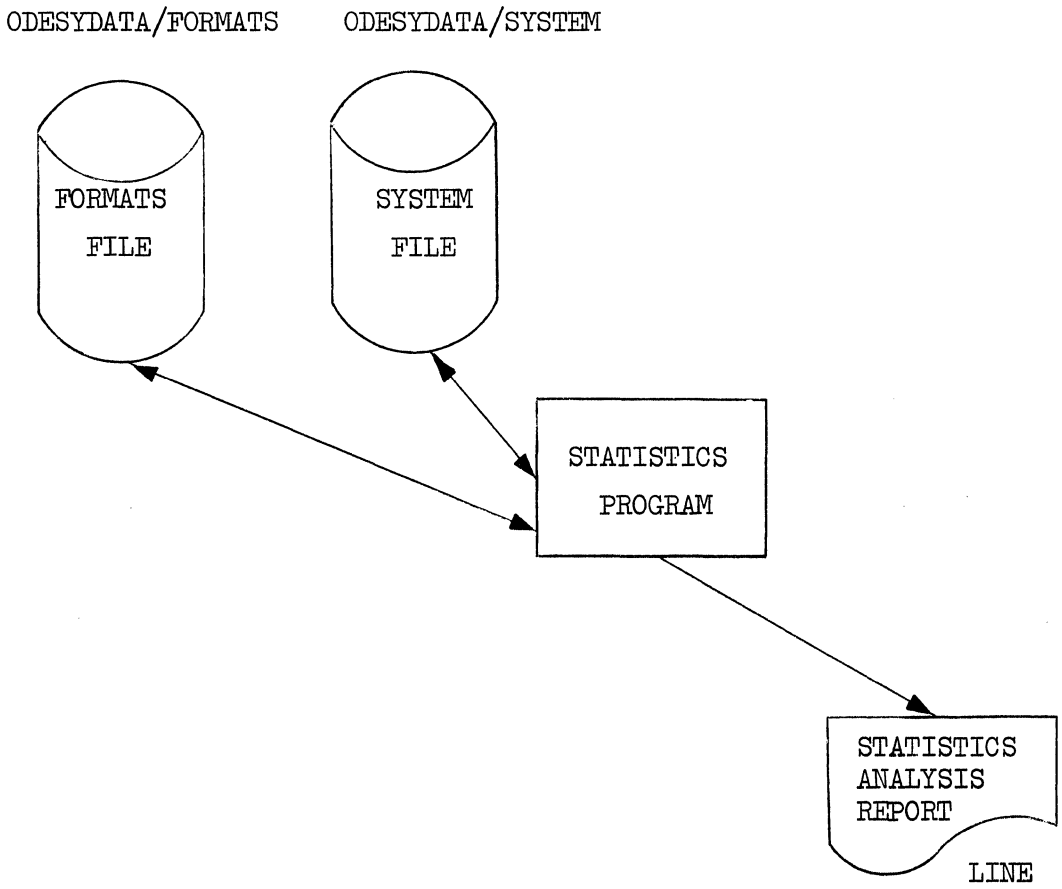
The operator report is more detailed. It shows the number of records entered and verified, the number of key strokes for enter mode and verify mode, the elapsed time (the time spent logged in at a terminal), the number of records per hour and the number of keystrokes per hour. There is a detail line for each operator, and a grand total for all operators.

With reference to the operator report, it should be noted that the details of elapsed time are gathered by the Edit, whereas the other details are gathered by the Extraction Program, and only refer to batches that have been released or deleted. If there is a long delay between entering and releasing batches, then running the Statistics Analysis Program once a day will produce distorted figures for records per hour and keystrokes per hour. For this reason it is recommended that the Statistics Analysis

Program be run once a week or once a month.

At the top of each report there is a transaction number to identify it. Each time the Statistics Analysis Program is run, this number is incremented by one. The report also shows the transaction number of the last report, and the date on which it was produced. Appendices D4 and D5 give examples of statistics reports.

7.2 STATISTICS PROGRAM BLOCK DIAGRAM



8. THE BATCH STATUS PRINT PROGRAM (PRINT)

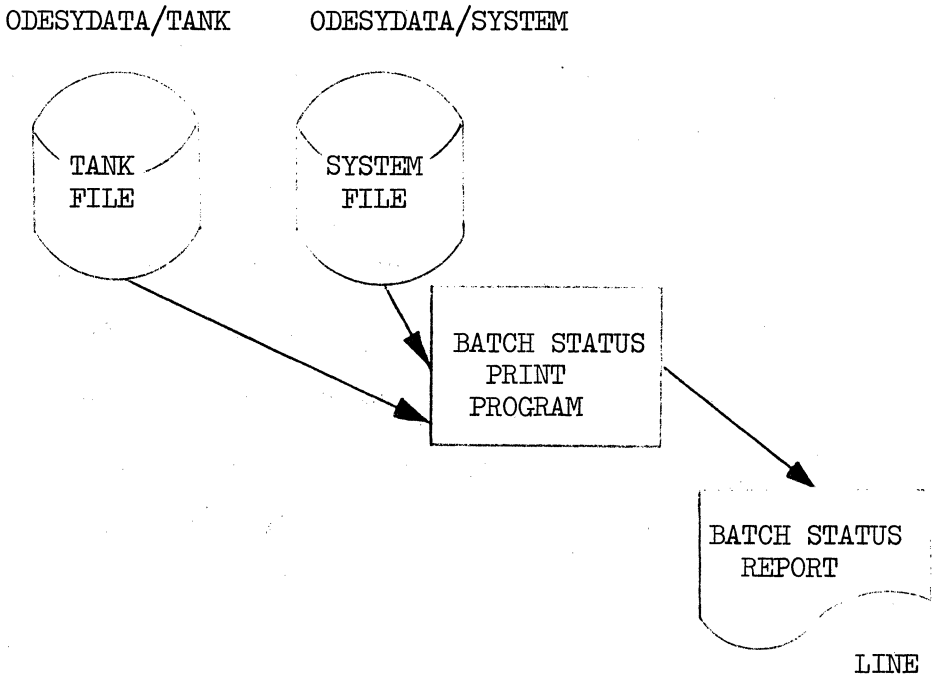
8.1 GENERAL

The Batch Status Print Program produces a report showing all existing batches and their current status.

For each batch, the report shows the operator who first opened the batch, the date it was first opened, the date it was last accessed, the operator who released the batch (if any), the number of records in the batch, the group associated with the batch, and the values of four status flags - enter status, verify status, extract status and delete status. As can be seen, the information is almost the same as is produced by the batch status command (See P.5-37).

It may be useful to run the Batch Status Print Program before running the Extract Program in order to ensure that the correct batches are scheduled for release or deletion.

8.2 BATCH STATUS PRINT PROGRAM BLOCK DIAGRAM



9. AUXILIARY PROGRAMS

9.1 GENERAL

There are two auxiliary ODESY programs. They are the Tank File Analysis Program (DUMP) and the Dump file Conversion Program (CONVERT). The first program is a debugging aid that may be used by your Burroughs System Support Representative. The second program is a conversion aid.

9.2 TANK FILE ANALYSIS PROGRAM (DUMP)

If a fatal error occurs during data entry, the EDIT program gives error message 74, PLEASE EXECUTE ODESY/DUMP. The Dump program produces a diagnostic report which should help in identifying the source of the trouble. The report should be sent to your local Burroughs System Support Representative together with the error report produced by the EDIT at the time of the error.

Operating Instructions

Enter from the SPO:

EX ODESY/DUMP

The program will produce a Tank File Analysis Report.

9.3 DUMPFILE CONVERSION PROGRAM (CONVERT)

The structure of the Formats file is liable to change between major releases of the programs, as extra facilities are made available. For this reason, a version of the Convert program is provided with each release to enable users to convert their existing formats. Once conversion is complete, the program may be removed from disk.

9.3 (Cont'd)

The program does not directly convert the Formats File. Instead it converts dump files (refer to /LOAD and /DUMP commands in Section 4). Therefore the user must create a dumpfile using the old version of Formaint, and after running the Convert program, the modified dumpfile must be reloaded to the Formats file using the new version of Formaint.

Operating Instructions

Enter from the SPO:

EX ODESY/CONVERT

The program will request the name of a dumpfile via a SPO accept message. Enter from the SPO:

<mix> AX <dumpfile name>

The program will modify this file such that it is acceptable to the new version of Formaint. It also produces a Dumpfile Conversion Report, showing all groups and formats that have been modified. ODESY/CONVERT will then request another dumpfile name.

The program may be terminated by the input of a SPO accept message containing blanks:

<mix> AX

10. INQUIRY FACILITIES

10.1 GENERAL

A number of inquiry facilities are available in ODESY. These inquiry facilities require the presence of a program (or programs) written by the user.

At run time, the user program communicates with the EDIT program to supply or check data.

The system is very simple in concept, but it allows:

(a) Inquiry on a data base;
(b) Updating of a data base;
(c) Existence checking;
(d) Interactive data entry;
(e) Inter-field checks.

The user program has the responsibility for accessing the data base, and for performing any checks not specified in the formats. A user program may be written in COBOL, RPG or UPL.

The flexibility of the inquiry facilities provided allows a very sophisticated user program to be written where this is necessary. On the other hand, a user program to perform existence checks or simple inquiries can be written very easily and quickly. An example of a simple user program to perform existence checks is shown in Worked Example 1.

If the inquiry facilities are not required, the user may ignore all references to them. This will not effect the running of the remainder of ODESY. The system parameter "maximum number of user programs" may be

10.1 (Cont'd)

set to zero.

In order to understand how the inquiry facilities work, it is necessary to examine the sequence of events during the entry of a record of data.

During data entry, an operator selects a format, enters data, and then selects another format. The EDIT program supplies default data before entry and performs checks after entry. The default data and the checks to be performed are specified in the format.

Similarly, a user program may supply data before entry and perform checks after entry. A user program will be involved whenever an inquiry format is selected (format type = K, S, C or U).

For a key data format (format type = K), the data entered is sent to the user program after the standard checks have been performed by the EDIT. For a supply data format (format type = S), the user program is asked to supply data before entry. For a check data format (format type = C), the data entered is checked by the EDIT, and then sent to the user program for further checking. A check-and-supply format (format type = U) combines the functions of the supply data and check data formats.

10.1 (Cont'd)

The following is the sequence of events by format type:

		blank or B	K	S	C	U
(a)	Select format	X	X	X	X	X
(b)	EDIT => User program			X		X
(c)	User program => EDIT			X		X
(d)	EDIT displays format	X	X	X	X	X
(e)	Operator enters data	X	X	X	X	X
(f)	EDIT checks data	X	X	X	X	X
(g)	EDIT => User program		X		X	X
(h)	User program => EDIT				X	X
(i)	EDIT stores data	X	(X)	(X)	(X)	(X)
(j)	Select next format	X	X	X	X	X

NOTES :

(1) If the EDIT finds errors at step (f) the format will be redisplayed with the errors indicated, after which step (e) will be repeated.

(2) A format may be selected in two ways: it may be selected by the operator (see enter command, // <format>), or it may be selected automatically when a format is specified with a FOLLOWING FORMAT.

(3) If inquiry mode is being used, there is no batch open, so that step (i) is omitted.

10.2 USER PROGRAM INTERFACE

Design considerations

There are only a few rules to be followed when writing a user program, and these apply only to the communication protocol.

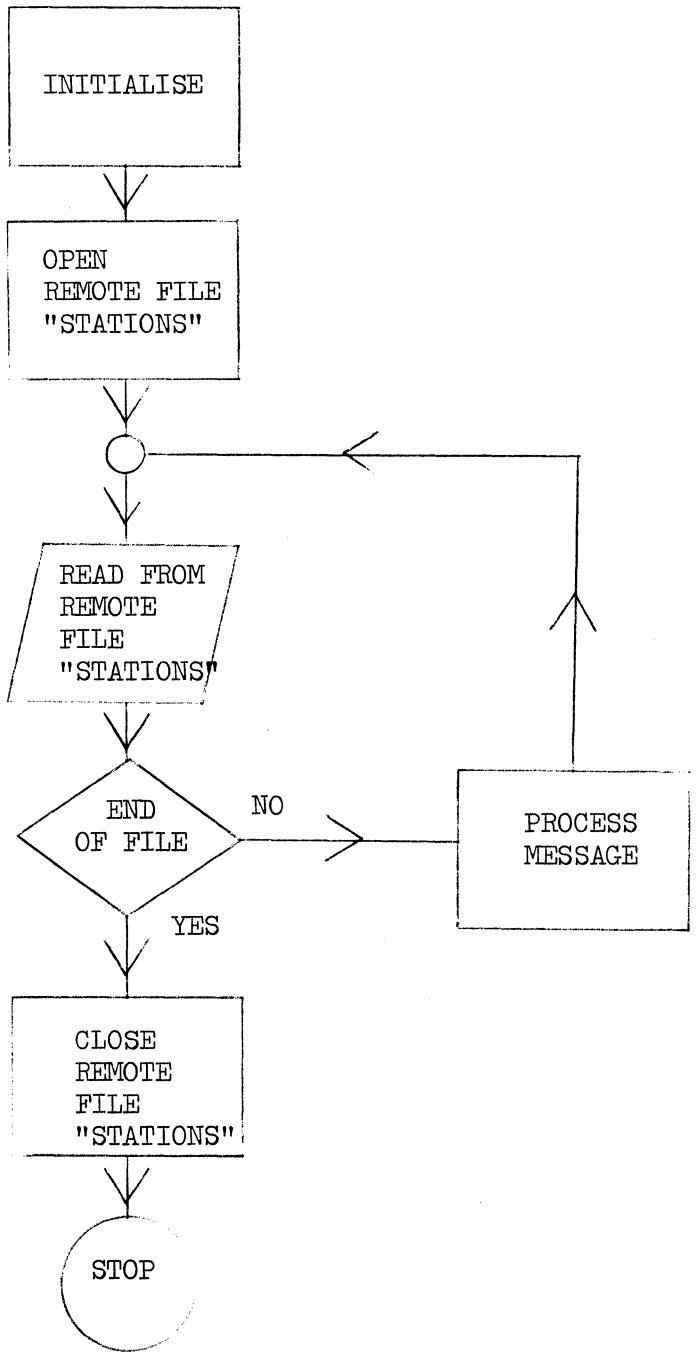
A user program may be written in COBOL, RPG or UPL. If COBOL is chosen, then DMSII constructs may be used to access a DMSII data base. Similarly, other data bases may be accessed by the user program.

It is not necessary for the console operator to execute a user program at run time. A user program will be ZIP-executed by the EDIT whenever it is required.

A user program communicates with the EDIT through the remote file STATIONS. A remote file is a method of queuing messages. The EDIT is an MCS-type program, and it uses the remote file to communicate with both terminals and user programs. Messages from a user program do not go directly to the terminals, they always go to the EDIT. (The EDIT sets the PARTICIPATING option when approving a file open). If the user program specifies a remote key, it will be ignored.

10.2 (Cont'd)

The structure of a user program will normally be the following:-



10.2 (Cont'd)

The main part of the program is a READ statement, followed by a set of procedure calls to process the message received, followed by a return to the READ statement (see Worked Examples 1 and 2). The READ statement should have an AT END or ON.EOF branch. This is because the EDIT causes an end of file condition when it wishes to terminate a user program.

The format of the messages in the remote file is described below.

Message Format

There is a standard format for all messages received and sent (READ and WRITE) by the user program. The message consists of a 30 byte header, which may be followed by data. In COBOL nomenclature the message is:

```
01  ODESY-RECORD-AREA.
    03  ODESY-MSG-TYPE              PIC   99.
    03  ODESY-STATION               PIC  999.
    03  ODESY-OPERATOR              PIC   99.
    03  ODESY-BATCH                 PIC  999.
    03  ODESY-RECORD                PIC 9999.
    03  ODESY-GROUP                 PIC XXXX.
    03  ODESY-FORMAT                PIC XXXX.
    03  ODESY-LENGTH                PIC 9999.
    03  ODESY-ERRORS                PIC   99.
    03  FILLER                      PIC   XX.

    03  ODESY-DATA                  PIC X(INPUT-DATA-SIZE).
```

10.2 (Cont'd)

The user program will normally receive a message and then send a message in reply. Those parameters which are not to be changed in the reply are noted below.

Message Type

Messages with the following message types are received by the user program from the EDIT:

01 key data message
02 supply data message
03 check data message

Messages with the following message types are expected to be sent by the user program to the EDIT:

11 data message
12 invalid key message
13 invalid format message

Station

This is the logical station number (LSN) of the station that initiated the message. It must not be changed by the user program.

Operator

This is the operator number of the operator logged in at the station. It must not be changed by the user program.

10.2 (Cont'd)

Batch

This is the batch number of the batch that is open at the station. If inquiry mode is being used, there is no batch open, and this parameter will be zero. It must not be changed by the user program.

Record

If a batch is open, this is the relative record number of the record being entered, modified, verified or inserted. It must not be changed by the user program.

Group

This is the group name used in the open command. It must not be changed by the user program.

Format

This is the current format being used at the station. It may only be changed in reply to a supply data message. Worked Example 2 shows how this may be done.

Length

This describes the length of ODESY-DATA. For an incoming supply data message, it is zero; for an incoming key data or check data message, it indicates the length of the data which follows the 30 byte header. In sending a data message, the user program is expected to set this parameter correctly.

10.2 (Cont'd)

Errors

In incoming messages, this is always zero. In replying to a check data message, the user program is expected to set this parameter if errors are found.

Data

In incoming keydata and check data messages, this contains the whole of the data for the record, in the order in which it was entered by the operator, that is in the order in which it was specified in the format. It will not be in the form expected by the output file (or application file). In outgoing data messages, the same applies. If the user program is replying to a check data message, and errors were detected, the fields in error should be marked (see Worked Examples 1 and 2).

The length of ODESY-DATA will never be greater than the value of the system parameter INPUT DATA SIZE. However, in any specific message, the actual length of ODESY-DATA will be given by ODESY-LENGTH.

Processing of incoming messages

The action taken by a user program upon receiving a message usually depends upon the group and format names, and the message type. However, the only rules laid down by ODESY refer to the message types of the messages sent in reply.

10.2 (Cont'd)

Key data message (message type = 01)

This is sent by the EDIT after successful checking of data entered for a key data format. If the EDIT finds errors in the data entered, this message is not sent until the errors have been corrected. The message contains the data entered. The user program must not send a message in reply to a key data message.

Supply data message (message type = 02)

This is sent by the EDIT after selection of a supply data format or a check-and-supply format. The message contains no data, but the user program is expected to send data in reply. The user program may send a data message (message type = 11) or an invalid key message (message type = 12) or an invalid format message (message type = 13). If a data message is sent, the format name may be changed.

Check data message (message type = 03)

This is sent by the EDIT after successful checking of data entered for a check data format or a check-and-supply format. If the EDIT finds errors in the data entered, this message is not sent until the errors have been corrected. The user program may send a data message (message type = 11) or an invalid key message (message type = 12) or an invalid format message (message type = 13). If a data message is sent, the ODESY-ERRORS parameter indicates whether the user program found errors in the data.

10.2 (Cont'd)

Outgoing messages

The normal reply to a supply data message or a check data message is a data message (message type = 11). The data message either supplies data to the EDIT, or returns the result of the checking of data. In both cases, the EDIT will normally present to the operator a format plus data.

However if the user program sends an invalid key message (message type = 12) or an invalid format message (message type = 13), then the EDIT will send a one-line error message to the operator. In the first case, it will send INVALID INQUIRY KEY (error message 66), and in the second case it will send INVALID INQUIRY FORMAT (error message 67). If the user program sends an unrecognised message to the EDIT, then error message 68, INVALID INQUIRY REPLY, will be sent to the operator.

10.3 INQUIRY FORMATS

There are four inquiry format types: key data, supply data, check data and check-and-supply. When FORMAINT is being used to enter an inquiry format, a user program name is required. This is the name of the user program that will receive all messages for this format. A user program must be able to handle messages for all inquiry formats for which it is named. An inquiry format cannot be used if the relevant user program is not on disk.

Keydata format

When a key data format is used, a key data message (message type = 01) is sent to the user program after successful checking of the data entered. The user program must not reply to this message.

10.3 (Cont'd)

The keydata format may be used for passing a key to the user program, or for passing data to the user program for which no response is required. A key data format would normally have a supply data format as its FOLLOWING FORMAT, so that the information supplied for the latter relates to the key entered for the former. If this method is to be used for inquiry, then the user program should have a table, with one entry for each station. The key supplied would be stored in this table, for use with later supply data or check data messages. Format AP VEND in Worked Example 2 is an example of a key data format.

Supply data format

When a supply data format is used, a supply data message (message type = 02) is sent to the user program before the operator is allowed to enter any data. The user program is expected to reply with a data message (message type = 11), or an invalid key message (message type = 12), or an invalid format message (message type = 13). If it is a data message, the EDIT inspects the format name in the message, and if this format is a supply data format or a check-and-supply format, the data at the end of the message is presented to the operator. The ability to change the format name allows the user program to alter the sequence of formats when there is no more data for this format or for this key. (See Worked Example 2, where the format is changed from AP INVC to AP VEND when there are no more invoices to supply).

A supply data format may be used for pure inquiry (no batch), or for interactive data entry (where the data supplied will help the operator in entering a batch). Format AP INVC in Worked Example 2 is an example of a supply data format.

10.3 (Cont'd)

Check data format

When a check data format is used, a check data message (message type = 03) is sent to the user program after successful checking by the EDIT of the data entered. The user program is expected to reply with a data message (message type = 11) or an invalid key message (message type = 12) or an invalid format message (message type = 13). If it is a data message, then the EDIT examines the ODESY-ERRORS parameter in the message. If this parameter is greater than zero, then the data in the message is presented to the operator for correction. If the ODESY-ERRORS parameter is zero, and there is a batch open (i.e not inquiry mode), then the data in the message is stored in the batch. Thus the user program could modify the data before returning it to the EDIT, and this modified data would be stored in the batch.

A check data format may be used for on-line updating (no batch) or for inter-field checks (where the allowable values for one field depend upon the value entered for another field), or for existence checking. Format INVC REC1 in Worked Example 1 is an example of a check data format being used for existence checking.

Check-and-supply format

When a check-and-supply format is used, a supply data message (message type = 02) is sent to the user program before the operator is allowed to enter any data, and a check data message (message type = 03) is sent to the user program after successful checking by the EDIT of the data entered.

A check-and-supply format combines the functions of a supply data format and a check data format. Format AP NAME in Worked Example 2 is an example of a check-and-supply format.

10.4 INQUIRY MODE

When inquiry formats are being used, batches are often unnecessary. Examples of this would be pure inquiry (i.e no data entry), and on-line updating (where the user program uses the data entered to directly update a data base). In such cases, inquiry mode may be used.

To get into inquiry mode, the operator keys in an open command followed by the keyword INQUIRY :

/OPEN INQUIRY <group>

The operator may now use the inquiry command to call up any format in the specified group:

//<format>

With the aid of the FOLLOWING FORMAT feature, the operator may be guided through an inquiry or update sequence. The AP formats in Worked Example 2 may be referenced in inquiry mode.

To get out of inquiry mode, the operator enters a close command:

/CLOSE

The user should note that the data entry commands enter, modify, verify, display, insert, delete and accept may not be used in inquiry mode.

10.5 USE OF DATA ENTRY COMMANDS WITH INQUIRY FORMATS

Accept command (/ACCEPT)

There is no difference between inquiry formats and non-inquiry formats. The record being entered or inserted is added to the batch, even though it contains errors. However, no message is sent to the user program to indicate that the record has been accepted.

Delete command (/DELETE)

There is no difference between inquiry formats and non-inquiry formats. The record is deleted from the batch. However, no message is sent to the user program to indicate that the record has been deleted.

Display command (/<record number>)

There is no difference between inquiry formats and non-inquiry formats. The data to be displayed will always be taken from the batch, even if it is a supply data format or a check-and-supply format. No message will be sent to the user program.

Enter command (//<format> in ENTER mode)

For a supply data format or a check-and-supply format, the user program will be asked to supply data before entry. For a keydata format or a check data format or a check-and-supply format, the data will be passed to the user program after entry.

10.5 (Cont'd)

Insert command (/INSERT)

For a supply data format or a check-and-supply format, the user program will be asked to supply data before entry. For a key data format or a check data format or a check-and-supply format, the data will be passed to the user program after entry.

Modify command (//<record number>in ENTER mode)

For a supply data format or a check-and-supply format, the data presented for modification will be taken from the batch. For a keydata format or a check data format or a check-and-supply format, the data will be sent to the user program after entry. Thus key data and check data messages are sent, but supply data messages are not sent.

Verify command (//<record number>in VERIFY mode)

For a supply data format or a check-and-supply format, the data presented for verification will be taken from the batch if the display verify option is set, or from the user program if the standard verify option is set. If the format is a key data format or a check data format or a check-and-supply format, the data will be sent to the user program after entry. Thus key data and check data messages are sent, and supply data messages are sent if standard verify is specified for the format.

10.6 WORKED EXAMPLE 1

Worked Example 1 demonstrates how a simple existence check may be performed.

Format INVCREC1 is for the entry of invoices, and the stock number in each invoice is to be checked by the user program STOCKINQ against a file of valid stock numbers. For simplicity, the stock number is used directly as a key to access the stock file. However the interface to ODESY would have remained the same if a data base were being accessed.

Listings of the format INVCREC1 and the program STOCKINQ may be found on pages 10-19 to 10-23.

Format INVCREC1

The format is a check data format, and refers to the user program STOCKINQ. The stock number appears in bytes 6 - 15 of the input record.

User program - data division

The record area for the remote file is first of all defined as 30 bytes of header information followed by 160 bytes of data. The 160 bytes of data is consistent with the Tank File Record Size option being set to A. The record area is then redefined so that it corresponds with the INVCREC1 format. The only important item in the redefinition is the stock number, which is to be checked.

10.6 (Cont'd)

User program - procedure division

During initialisation, the stock file and the remote file are opened. The rest of the program is a loop to read the remote file and process the messages received.

If the format in the message is not INVCREC1, an invalid format message (message type = 13) is sent. Otherwise the stock number is checked against the stock file. If the number is valid, a data message (message type = 11) is sent, with the data unchanged and ODESY-ERRORS = 0. If the stock number is invalid a data message is still sent, but asterisks are used to indicate a stock number error in the data, and ODESY-ERRORS = 1.

On end of file (for the remote file), the program terminates.

```
B1700 DATA ENTRY SYSTEM      F O R M A T   M A I N T E N A N C E   R E P O R T      TIME 11:58  DATE 08/03/77  PAGE   1

   GROUP NAME  INVC                          FILE NAME          IC110                  DATE LAST CHANGED 25/02/77

   RECORD SIZE  80   BLOCKED  1   RECORDS/AREA  500   AREAS 40      ON DISK                              REPORT


   FORMAT NAME INVCREC1          NEXT FORMAT INVCREC1   SECURITY  50     D-VERIFY   CHECK     DATE LAST CHANGED 27/02/77
                                                                          USER PROGRAM NAME            STOCKING

SEQ    NAME    LGTH TYP DUP END CNS/VAL XFT CDV BTCT MAND JUST   CRT WRT DUP BLNK PACK JUST SIGN SIGN   PRINT PRINT PRINT
                    IN  LINE /RANGE                     IN    POS AFT OUT FILL      CLT      CONV   COL   AFTER NAME
 10 BTCH.NO2/3   5  N   D   E                          M         3          B          R              23
 20 STK.NO.     15  N       E                          M    R    8          B          R              29
 30 ST.CODE      1              V                      M        23                                    45          P
                  CONSTANTS    A S U
 40 TR.NO.       5  N       E                          M        24          B          R              56
 50 QTY.         9  N                             1          L  29          B          R              62
 60 D.F          1          E     V                   M        38                                    72
                  CONSTANTS    0 1
 70 VALUE       11  N       E                          2        39          B          R              74          P
 80 DATE         6  N   D                              M        50                                    92
 90 REC.CODE     1  N       E     V                             1                                     1          P
                  CONSTANTS    2 3
100 AUX CODE     1  N             V                             2  C                                  12     C    P
                  CONSTANTS      0 1 2

**** TOTALS ****          FIELDS INPUT   10          FIELDS OUTPUT  10          FIELDS PRINTED   10
                          INPUT DATA SIZE   55       CHARACTERS OF CONSTANT DATA   11
```

Figure 10.1 Format INVCREC1

```
000100 /
000200   IDENTIFICATION DIVISION.
000300 * ------------------------
000400   PROGRAM-ID. STOCKINQ.
000500   AUTHOR. DATA ENTRY GROUP.
000600   DATE-WRITTEN. FEBRUARY 1977.
000700   DATE-COMPILED.
                      TIME 11:30 DATE 03/08/77.
000800 *
000900   ENVIRONMENT DIVISION.
001000 * ---------------------
001100   CONFIGURATION SECTION.
001200   SOURCE-COMPUTER. B-1700.
001300   OBJECT-COMPUTER. B-1700.
001400   INPUT-OUTPUT SECTION.
001500   FILE-CONTROL.
001600       SELECT STOCK-FILE ASSIGN TO DISK,
001700              ACCESS MODE IS RANDOM,
001800              ACTUAL KEY IS STOCK-KEY.
001900       SELECT REMOTE-FILE ASSIGN TO REMOTE.
```


Figure 10.2 STOCKINQ (1 of 4)

```
002000 /
002100   DATA DIVISION.
002200 * -------------
002300   FILE SECTION.
002400   FD  STOCK-FILE
002500       BLOCK CONTAINS 20 RECORDS
002600       VALUE OF ID IS "STOCKFILE".
WARNING...(222) MISSING FILE CONTAINS CLAUSE - 1 AREA OF 500 BLOCKS BY DEFAULT
002700   01  STOCK-REC.                                                    000^0000056
002800       03 STOCK-NO          PIC 9(8).                               000^0000056
002900       03 STOCK-VALID       PIC 9.
003000 *                                                                  000^0000072
003100   FD  REMOTE-FILE
003200       VALUE OF ID IS "STATIONS".
003300   01  CDESY-RECORD-AREA.                                            000^0000074
003400       03 CDESY-MSG-TYPE    PIC 99.                          [0001] 000^0000074
003500       03 CDESY-STATION     PIC 999.                                000^0000078
003600       03 CDESY-OPERATOR    PIC 99.                                 000^0000084
003700       03 CDESY-BATCH       PIC 999.                                000^0000088
003800       03 CDESY-RECORD      PIC 9999.                               000^0000094
003900       03 CDESY-GROUP       PIC XXXX.                               000^0000102
004000       03 CDESY-FORMAT      PIC XXXX.                               000^0000110
004100       03 CDESY-LENGTH      PIC 9999.                               000^0000118
004200       03 CDESY-ERRORS      PIC 99.                                 000^0000126
004300       03 FILLER            PIC XX.                                 000^0000130
004400       03 CDESY-DATA        PIC X(160).                             000^0000134
004500   01  INVCREC1-AREA.                                                000^0000074
004600       03 FILLER            PIC X(30).                              000^0000074
004700       03 INVC-BATCH        PIC 9(5).                               000^0000134
004800       03 INVC-STOCK-NO     PIC 9(15).                              000^0000144
004900       03 INVC-ST-NO-ERRORS REDEFINES INVC-STOCK-NO PIC X(15).      000^0000144
005000       03 INVC-STOCK-CODE   PIC X.                                  000^0000174
005100       03 INVC-TR-NO        PIC 9(5).                               000^0000176
005200       03 INVC-QUANTITY     PIC 9(9).                               000^0000186
005300       03 INVC-DF           PIC X.                                  000^0000204
005400       03 INVC-VALUE        PIC 9(11).                              000^0000206
005500       03 INVC-DATE         PIC 9(6).                               000^0000228
005600       03 INVC-REC-CODE     PIC 9.                                  000^0000240
005700       03 INVC-AUX-CODE     PIC 9.
005800 *                                                                  000^0000242
005900   WORKING-STORAGE SECTION.
006000   77  STOCK-KEY            PIC 9(8) COMP.
```


Figure 10.3 STOCKINQ (2 of 4)

```
006100 /                                                                  000,0000454
006200   PROCEDURE DIVISION.
006300 * ------------------                                               000,0000000
006400   INITIALISE.                                                      000,0000000
006500       OPEN INPUT STOCK-FILE.                                       000,0000000
006600       OPEN I-O REMOTE-FILE.                                        000,0000080
006700   MAIN-LOOP.                                                       000,0000160
006800       READ REMOTE-FILE AT END STOP RUN.                            000,0000160
006900       IF ODESY-GROUP = "INVC" AND ODESY-FORMAT = "REC1"            000,0000414
007000          GO TO INVCREC1-CHECK.                                     000,0000493
007100 *     ELSE INVALID FORMAT                                          000,0000587
007200       MOVE 13 TO ODESY-MSG-TYPE.                                   000,0000587
007300   WRITE-REMOTE.                                                    000,0000607
007400       WRITE ODESY-RECORD-AREA.                                     000,0000607
007500       GO TO MAIN-LOOP.                                             000,0000759
007600 *                                                                  000,0000774
007700   INVCREC1-CHECK.                                                  000,0000774
007800       MOVE INVC-STOCK-NO TO STOCK-KEY.                             000,0000774
WARNING...(252) RECEIVING FIELD TRUNCATION
007900       READ STOCK-FILE INVALID KEY GO TO INVCREC1-ERRER.            000,0000824
008000       IF STOCK-VALID = 0 GO TO INVCREC1-ERROR.                     000,0001099
008100       MOVE 11 TO ODESY-MSG-TYPE.                                   000,0001146
008200       GO TO WRITE-REMOTE.                                          000,0001166
008300   INVCREC1-ERROR.                                                  000,0001181
008400       MOVE "***************" TO INVC-ST-NO-ERRORS.                 000,0001181
008500       MOVE 1 TO ODESY-ERRORS.                                      000,0001341
008600       MOVE 11 TO ODESY-MSG-TYPE.                                   000,0001377
008700       GO TO WRITE-REMOTE.                                          000,0001397
008800   END-OF-JOB.                                                      000,0001412
```


Figure 10.4 STOCKINQ (3 of 4)

```
  C  O  D  E     D  I  C  T  I  O  N  A  R  Y
      BYTE LENGTH    CODEFILE RELATIVE DISK ADR
      ...........    ..........................
  000 00000177       000006
      00000177       TOTAL CODE

  D  A  T  A     D  I  C  T  I  O  N  A  R  Y
      BYTE LENGTH    CODEFILE RELATIVE DISK ADR
      ...........    ..........................
  000 00000283       000003

  S-M A C H I N E    P A R A M E T E R S    (S C R A T C H P A D)

  LENB=6, SEGB=0, DISPB=10, CCPXB=3, COPB=20, D.E.F=254, BDISPB=11
  BASE RELATIVE ADDRESSES:
    DATA-SEGMENT-0=204, COP-TABLE=44

  P R O G R A M    P A R A M E T E R    B L O C K

  FIRST-EXECUTABLE-INSTRUCTION=0,0
  INTERPRETER-NAME=COBCL      /INTERP
  STATIC-CORE=2264 BITS
  DYNAMIC-CORE=0 BITS
  DATA DICTIONARY STARTS AT CODEFILE SEGMENT 2, 1 ENTRY
  CODE DICTIONARY STARTS AT CODEFILE SEGMENT 5, 1 ENTRY
  FILE PARAMETER BLOCKS START AT CODEFILE SEGMENT 7, 2 ENTRIES

LAST ERROR AT SEQUENCE NUMBER       . 2 WARNINGS
***** COMPILATION COMPLETE
ELAPSED TIME : 03 MINUTES, 05 SECONDS
PROGRAM REQUIRES 9 DISK SEGMENTS OF 180 BYTES EACH.
MEMORY REQUIREMENTS
  0000177 BYTES = LARGEST CODE SEGMENT
  0000283 BYTES = BASE-TO-LIMIT AREA
  0000315 BYTES = DICTIONARIES AND RUN STRUCTURE
  0000734 BYTES = FILE BUFFERS & FILE INFO AREAS - INCLUDES 129 BYTES (+72 TO 540 IF DISK) FOR EACH FILE
  0001509 BYTES = ESTIMATED MEMORY REQUIRED TO RUN IF ALL FILES OPEN
89 SYMBOLIC RECORDS COMPILED AT 28.860 RECORDS PER MINUTE.
```


Figure 10.5 STOCKINQ (4 of 4)

10.7 WORKED EXAMPLE 2

Worked Example 2 demonstrates some of the more sophisticated inquiry facilities offered by ODESY.

It shows an operator examining selected records from a vendor master file, and then scanning the vendor's outstanding invoices, where this is necessary.

Listings of the formats APVEND, APNAME and APINVC, and the user program APINQ may be found on pages 10-28 to 10-37.

Formats APVEND, APNAME and APINVC

Format APVEND is a key data format, format APNAME is a check-and-supply format, and format APINVC is a supply data format. All formats refer to the user program APINQ.

User program - data division

There are record areas for the vendor master file, the invoice file and the remote file. The record area for the remote file is redefined so that it corresponds with the data for the three formats. There is also a station table that is used to store keys for the vendor master file and the invoice file.

User program - procedure division

During initialisation, the vendor master file, the invoice file and the remote file are opened, and the station table is set to zeros. The rest of the program is a loop to read the remote file and process the messages received.

10.7 (Cont'd)

If the message is a keydata message for format APVEND, the vendor number is stored in the table as THIS-VEND and NEXT-VEND. No message is sent in reply.

If the message is a supply data message for format APNAME, then NEXT-VEND is examined. If it contains a non-zero value, the vendor record is found, and the relevant information sent to the EDIT in a data message. Format APNAME requires 100 characters of data. If NEXT-VEND is zero, the format name is changed to APINVC, and the first invoice in the invoice file for THIS-VEND is found. The relevant information is sent to the EDIT in a data message. Format APINVC requires 88 characters of data.

If the message is a check data message for format APNAME, the value in the NEXT VENDOR field is stored in NEXT-VEND. If the operator has not entered a number, NEXT VENDOR will be blank. A data message is sent in reply, with ODESY-ERRORS = 0.

If the message is a supply data message for format APINVC, the next invoice for THIS-VEND is found. The relevant information is sent to the EDIT in a data message. However, if there are no more invoices for THIS-VEND, the format name is changed to APVEND, and a data message (containing no data) is sent to the EDIT.

Operation

The operator may select inquiry mode, or alternatively a batch may be opened in enter mode to provide a journal. The journal would show all inquiries performed on the files.

10.7 (Cont'd)

After entering the /HELLO and /OPEN commands, the operator keys in:

//VEND

The APVEND format is displayed and the operator keys in a vendor number.

The APVEND data is sent to the user program in a key data message. APVEND is followed by APNAME, so that a supply data message is immediately sent to the user program. NEXT-VEND will be non-zero, so the vendor details are supplied in a data message.

The APNAME format is displayed at the terminal, together with the vendor details.

If the operator wishes to look at another vendor master record, then a value is entered for the field NEXT VENDOR. However, if the operator transmits without changing the data, the first invoice will be supplied.

In either case, the APNAME data is sent to the user program in a check data message. The user program stores NEXT VENDOR as NEXT-VEND in its table, and sends a data message in reply with ODESY-ERRORS = 0.

The APNAME format is followed by APNAME, so another supply data message is immediately sent to the user program. If NEXT-VEND is zero, the format name is changed to APINVC, and the first invoice details are sent to the EDIT in a data message.

10.7 (Cont'd)

The APINVC format is displayed at the terminal, together with the vendor details

If the operator transmits without changing the data, the next invoice will be supplied; alternatively by keying in:

//NAME

the vendor details may be recalled; and another vendor master record may be examined by keying in:

//VEND

and then the vendor number.

If the operator chooses to look at all the invoices, then the format will automatically change to APVEND after the last invoice.

```
B1700 DATA ENTRY SYSTEM      F O R M A T   M A I N T E N A N C E   R E P O R T      TIME 11:57  DATE 08/03/77  PAGE   1

   GROUP NAME  AF                              FILE NAME          NONE                    DATE LAST CHANGED 08/03/77

   RECORD SIZE 180   BLOCKED  1   RECORDS/AREA  500   AREAS 40      ON DISK                                REPORT


   FORMAT NAME AF  VEND          NEXT FORMAT AP  NAME   SECURITY  80      VERIFY   NDATA     DATE LAST CHANGED 27/02/77
                                                                          USER PROGRAM NAME            APINQ

SEQ    NAME   LGTH TYP DUP END CNS/VAL XFT CDV BTCT MAND JUST   CRT WRT DUP BLNK PACK JUST SIGN SIGN   PRINT PRINT PRINT
                 IN  LINE /RANGE                      IN    POS AFT OUT FILL       CLT      CONV   COL   AFTER NAME
 10 VENDOR NUM   6  N                                M    R
**** WARNING ****  THIS FIELD IS NEITHER OUTPUT NOR PRINTED

**** WARNING ****  THE BUFFER IS NOT WRITTEN AFTER LAST FIELD
**** TOTALS  ****           FIELDS INPUT    1            FIELDS OUTPUT   0          FIELDS PRINTED    0
                           INPUT DATA SIZE    6         CHARACTERS OF CONSTANT DATA    0
```

Figure 10.6 Format AP VEND

B1700 DATA ENTRY SYSTEM F O R M A T M A I N T E N A N C E R E P O R T TIME 11:57 DATE 08/03/77 PAGE 2

FORMAT NAME AP NAME NEXT FORMAT AP NAME SECURITY 80 VERIFY C-8-S DATE LAST CHANGED 08/03/77
USER PROGRAM NAME APINQ

SEQ	NAME	LGTH	TYP	CUP IN	END LINE	CNS/VAL /RANGE	XFT	CDV	BTCT	MAND	JUST IN	CRT FCS	BRT AFT	DUP OUT	BLNK FILL	PACK	JUST CLT	SIGN	SIGN CONV	PRINT COL	PRINT AFTER	PRINT NAME
10	VENDOR NUM	6	N		E															1		P
20		24			E															20		
30		24			E															50		
40		24			E															80	C	
50	PAYBLE BAL	10	N		E															20		P
60	LST ACTIVE	6	N		E															50		P
70	NEXT VEND	6	N								R										C	

**** WARNING **** THIS FIELD IS NEITHER OUTPUT NOR PRINTED

**** WARNING **** THE BUFFER IS NOT WRITTEN AFTER LAST FIELD

**** TOTALS **** FIELDS INPUT 7 FIELDS OUTPUT 0 FIELDS PRINTED 6
INPUT DATA SIZE 100 CHARACTERS OF CONSTANT DATA 0

Figure 10.7 Format AP NAME

```
B1700 DATA ENTRY SYSTEM      F O R M A T   M A I N T E N A N C E   R E P O R T      TIME 11:57  DATE 08/03/77  PAGE   3

   FORMAT NAME AP  INVC          NEXT FORMAT AP  INVC   SECURITY  80       VERIFY   SUFF      DATE LAST CHANGED 08/03/77
                                                                          USER PROGRAM NAME            APINQ

SEQ    NAME    LGTH TYP DUP END CNS/VAL XFT CDV BTOT MAND JUST   CUT WRT DUP BLNK PACK JUST SIGN SIGN   PRINT PRINT PRINT
                    IN  LINE /RANGE                    IN    PES AFT OUT FILL       CLT      CONV   COL   AFTER NAME
 10 VENDOR NUM    6  N       E
**** WARNING ****  THIS FIELD IS NEITHER OUTPUT NOR PRINTED
 20 TRAN NUM      6  N                                                                              20
 30 INV NUM      10          E                                                                    30
 40 DATE          6  N                                                                           44         P
 50 DUE DATE      6  N       E                                                                    55    C    P
 60 AMOUNT       10  N       E                                                                    35         P
 70 PAID         10  N                                                                           56         P
 80 PAYABLE      10  N       E                                                                    75         P
 90              24                                                                             97    C

**** WARNING ****  THE BUFFER IS NOT WRITTEN AFTER LAST FIELD
**** TOTALS  ****          FIELDS INPUT    9            FIELDS OUTPUT   0            FIELDS PRINTED    8
                          INPUT DATA SIZE   88          CHARACTERS OF CONSTANT DATA    0
```


Figure 10.8 Format AP INVC

```
000100 /
000200   IDENTIFICATION DIVISION.
000300 * ------------------------
000400   PROGRAM-ID. APINQ.
000500   AUTHOR. DATA ENTRY GROUP.
000600   DATE-WRITTEN. FEBRUARY 1977.
000700   DATE-COMPILED.
                    TIME 11:27 DATE 03/08/77.
000800 *
000900   ENVIRONMENT DIVISION.
001000 * --------------------
001100   CONFIGURATION SECTION.
001200   SOURCE-COMPUTER. B-1700.
001300   OBJECT-COMPUTER. B-1700.
001400   INPUT-OUTPUT SECTION.
001500   FILE-CONTROL.
001600       SELECT VENDOR-MASTER ASSIGN TO DISK,
001700              ACCESS MODE IS RANDOM,
001800              ACTUAL KEY IS VENDOR-KEY.
001900       SELECT INVOICE-FILE ASSIGN TO DISK,
002000              ACCESS MODE IS RANDOM,
002100              ACTUAL KEY IS INVOICE-KEY.
002200       SELECT REMOTE-FILE ASSIGN TO REMOTE.
```


Figure 10.9 APINQ (1 of 7)

```
002300 /
002400   DATA DIVISION.
002500 * --------------
002600   FILE SECTION.
002700   FD  VENDOR-MASTER
002800       VALUE OF ID IS "APVNMR".
WARNING...(222) MISSING FILE CONTAINS CLAUSE - 1 AREA OF 500 BLOCKS BY DEFAULT
002900   01  VENDOR-REC-AREA.                                                    000,0000056
003000       03 VEND-VENDNO       PIC 9(6) COMP.                                000,0000056
003100       03 VEND-NAME         PIC X(24).                                    000,0000062
003200       03 VEND-ADDR         PIC X(24).                                    000,0000110
003300       03 VEND-ADDR2        PIC X(24).                                    000,0000158
003400       03 FILLER            PIC X(16).                                    000,0000206
003500       03 VEND-PAYBAL       PIC S9(9) COMP.                  [00001]      000,0000238
003600       03 FILLER            PIC X(35).                                    000,0000248
003700       03 VEND-DATE-LA      PIC 9(6).                                     000,0000318
003800       03 FILLER            PIC X(43).                                    000,0000330
003900   FD  INVOICE-FILE
004000       BLOCK CONTAINS 2 RECORDS
004100       VALUE OF ID IS "APVNIN".
WARNING...(222) MISSING FILE CONTAINS CLAUSE - 1 AREA OF 500 BLOCKS BY DEFAULT
004200   01  INVOICE-REC-AREA.                                                   000,0000416
004300       03 INVC-VENDNO       PIC 9(6) COMP.                   [00002]      000,0000416
004400       03 INVC-TRANNO       PIC 9(6) COMP.                                000,0000422
004500       03 INVC-INVNO        PIC X(10).                                    000,0000428
004600       03 INVC-DATE         PIC 9(6).                                     000,0000448
004700       03 INVC-DUE-DATE     PIC 9(6).                                     000,0000460
004800       03 INVC-AMOUNT       PIC S9(9) COMP.                  [00003]      000,0000472
004900       03 INVC-PAID         PIC S9(9) COMP.                  [00004]      000,0000482
005000       03 INVC-PAYABLE      PIC S9(9) COMP.                  [00005]      000,0000492
005100       03 FILLER            PIC X(19).                                    000,0000502
005200       03 INVC-DESC         PIC X(24).                                    000,0000540
005300       03 FILLER            PIC X(4).
```


Figure 10.10 APINQ (2 of 7)

```
005400 /                                                                    000,0000588
005500   FD  REMOTE-FILE
005600       VALUE OF ID IS "STATIONS".
005700   01  CDESY-RECORD-AREA.                                      [0006] 000,0000596
005800       03 CDESY-MSG-TYPE    PIC 99.                           [0007] 000,0000598
005900       03 CDESY-STATION     PIC 999.                          [0008] 000,0000600
006000       03 CDESY-OPERATOR    PIC 99.                                  000,0000606
006100       03 CDESY-BATCH       PIC 999.                                 000,0000610
006200       03 CDESY-RECORD      PIC 9999.                                000,0000616
006300       03 CDESY-GROUP       PIC XXXX.                                000,0000624
006400       03 CDESY-FORMAT      PIC XXXX.                         [0009] 000,0000632
006500       03 CDESY-LENGTH      PIC 9999.                         [0010] 000,0000640
006600       03 CDESY-ERRORS      PIC 99.                                  000,0000648
006700       03 FILLER            PIC XX.                                  000,0000652
006800       03 CDESY-DATA        PIC X(160).                       [0011] 000,0000656
006900   01  APVEND-AREA.                                                  000,0000596
007000       03 FILLER            PIC X(30).                               000,0000598
007100       03 APVEND-VENDNO     PIC 9(6).                         [0012] 000,0000656
007200   01  APNAME-AREA.                                                  000,0000598
007300       03 FILLER            PIC X(30).                               000,0000598
007400       03 APNAME-VENDNO     PIC 9(6).                         [0013] 000,0000656
007500       03 APNAME-NAME       PIC X(24).                        [0014] 000,0000668
007600       03 APNAME-ADDR       PIC X(24).                               000,0000716
007700       03 APNAME-ADDR2      PIC X(24).                               000,0000764
007800       03 APNAME-SPAYBAL    PIC X.                                   000,0000812
007900       03 APNAME-PAYBAL     PIC 9(9).                                000,0000814
008000       03 APNAME-DATE       PIC 9(6).                                000,0000832
008100       03 APNAME-NEXTVEND   PIC 9(6).                                000,0000844
008200   01  APINVC-AREA.                                                  000,0000596
008300       03 FILLER            PIC X(30).                               000,0000598
008400       03 APINVC-VENDNO     PIC 9(6).                                000,0000656
008500       03 APINVC-TRANNO     PIC 9(6).                                000,0000668
008600       03 APINVC-INVNO      PIC X(10).                               000,0000680
008700       03 APINVC-DATE       PIC 9(6).                                000,0000700
008800       03 APINVC-DUE-DATE   PIC 9(6).                                000,0000712
008900       03 APINVC-SAMCUNT    PIC X.                                   000,0000724
009000       03 APINVC-AMOUNT     PIC 9(9).                                000,0000726
009100       03 APINVC-SPAID      PIC X.                                   000,0000744
009200       03 APINVC-PAID       PIC 9(9).                                000,0000746
009300       03 APINVC-SPAYABLE   PIC X.                                   000,0000764
009400       03 APINVC-PAYABLE    PIC 9(9).                                000,0000766
009500       03 APINVC-DESC       PIC X(24).                               000,0000784
009600   WORKING-STORAGE SECTION.
009700   77  VENDOR-KEY           PIC 9(8) COMP.                           000,0000976
009800   77  INVOICE-KEY          PIC 9(8) COMP.                    [0015] 000,0000984
009900   77  I                    PIC 99 COMP.                      [0016] 000,0000992
010000   01  STATION-TABLE.                                                000,0000994
010100       03 STATION-ENTRY  OCCURS 10.                                  000,0000994
010200           05 THIS-VEND     PIC 9(6) COMP.                    [0017] 000,0000994
010300           05 NEXT-VEND     PIC 9(6) COMP.                    [0018] 000,0001000
010400           05 THIS-INV      PIC 9(8) COMP.
```


Figure 10.11 APINQ (3 of 7)

```
010500 /                                                                  [00211] 000,0001006
010600   PROCEDURE DIVISION.
010700 * ------------------                                                      000,0000000
010800   INITIALISE.                                                           000,0000000
010900       OPEN INPUT VENDOR-MASTER.                                         000,0000000
011000       OPEN INPUT INVOICE-FILE.                                          000,0000084
011100       OPEN I-O REMOTE-FILE.                                             000,0000168
011200       MOVE 1 TO I.                                                      000,0000252
011300   INIT-STATION-TABLE-LOOP.                                              000,0000270
011400       MOVE 0 TO THIS-VEND(I).                                           000,0000270
011500       MOVE 0 TO NEXT-VEND(I).                                           000,0000289
011600       MOVE 0 TO THIS-INV(I).                                            000,0000308
011700       ADD 1 TO I.                                                       000,0000327
011800       IF I < 11 GO TO INIT-STATION-TABLE-LOOP.                          000,0000341
011900   MAIN-LOOP.                                                            000,0000380
012000       READ REMOTE-FILE AT END STOP RUN.                                 000,0000380
012100       IF CDESY-GROUP = "AP" GO TO AP-CHECK.                             000,0000650
012200   INVALID-FORMAT.                                                       000,0000720
012300       MOVE 13 TO CDESY-MSG-TYPE.                                        000,0000720
012400       WRITE CDESY-RECORD-AREA.                                          000,0000742
012500       GO TO MAIN-LOOP.                                                  000,0000898
012600   INVALID-KEY.                                                          000,0000915
012700       MOVE 12 TO CDESY-MSG-TYPE.                                        000,0000915
012800       WRITE CDESY-RECORD-AREA.                                          000,0000937
012900       GO TO MAIN-LOOP.                                                  000,0001093
013000   WRITE-REMOTE.                                                         000,0001110
013100       MOVE 11 TO CDESY-MSG-TYPE.                                        000,0001110
013200       WRITE CDESY-RECORD-AREA.                                          000,0001132
013300       GO TO MAIN-LOOP.                                                  000,0001288
013400   AP-CHECK.                                                             000,0001305
013500       IF CDESY-FORMAT = "VEND" GO TO APVEND-KEYDATA.                    000,0001305
013600       IF THIS-VEND(CDESY-STATION) = 0 GO TO INVALID-KEY.                000,0001368
013700       IF CDESY-FORMAT = "NAME" AND CDESY-MSG-TYPE = 2                   000,0001404
013800          GO TO APNAME-SUPPLY.                                           000,0001467
013900       IF CDESY-FORMAT = "NAME" AND CDESY-MSG-TYPE = 1                   000,0001519
014000          GO TO APNAME-CHECK.                                            000,0001582
014100       IF CDESY-FORMAT = "INVC" GO TO APINVC-SUPPLY.                     000,0001634
014200       GO TO INVALID-FORMAT.                                             000,0001697
```


Figure 10.12 APINQ (4 of 7)

```
014300 /                                                                    000,0001714
014400   APVEND-KEYDATA.                                                    000,0001714
014500       MOVE APVEND-VENDNO TO THIS-VEND(ODESY-STATION).                000,0001714
014600       MOVE APVEND-VENDNO TO NEXT-VEND(ODESY-STATION).                000,0001733
014700       MOVE 0 TO THIS-INV(ODESY-STATION).                             000,0001752
014800       GO TO MAIN-LOOP.                                               000,0001771
014900 *                                                                    000,0001788
015000   APNAME-SUPPLY.                                                     000,0001788
015100       IF NEXT-VEND(ODESY-STATION) = 0                                000,0001788
015200 *        THEN SUPPLY FIRST INVOICE FOR THIS VENDOR                  000,0001788
015300           GO TO APNAME-FIRST-INVOICE.                                000,0001788
015400 *        ELSE SUPPLY NEXT VENDOR DETAILS                            000,0001824
015500       MOVE NEXT-VEND(ODESY-STATION) TO THIS-VEND(ODESY-STATION).      000,0001824
015600       MOVE THIS-VEND(ODESY-STATION) TO VENDOR-KEY.                   000,0001848
015700       READ VENDOR-MASTER INVALID KEY GO TO MISSING-VENDER.           000,0001890
015800       MOVE SPACES TO ODESY-DATA.                                     000,0002185
015900       MOVE VEND-VENDNO TO APNAME-VENDNO.                             000,0002199
016000       MOVE VEND-NAME TO APNAME-NAME.                                 000,0002236
016100       MOVE VEND-ADDR TO APNAME-ADDR.                                 000,0002273
016200       MOVE VEND-ADDR2 TO APNAME-ADDR2.                               000,0002333
016300       MOVE VEND-PAYBAL TO APNAME-PAYBAL                              000,0002393
016400       IF VEND-PAYBAL < 0 MOVE "-" TO APNAME-SPAYBAL.                 000,0002393
016500       MOVE VEND-DATE-LA TO APNAME-DATE.                              000,0002506
016600       MOVE 100 TO ODESY-LENGTH.                                      000,0002566
016700       GO TO WRITE-REMOTE.                                            000,0002592
016800   MISSING-VENDOR.                                                    000,0002609
016900       MOVE SPACES TO ODESY-DATA.                                     000,0002609
017000       MOVE THIS-VEND(ODESY-STATION) TO APNAME-VENDNO.                000,0002623
017100       MOVE "MISSING VENDOR" TO APNAME-NAME.                          000,0002642
017200       MOVE 100 TO ODESY-LENGTH.                                      000,0002776
017300       GO TO WRITE-REMOTE.                                            000,0002802
017400   APNAME-FIRST-INVOICE.                                              000,0002819
017500       MOVE THIS-VEND(ODESY-STATION) TO NEXT-VEND(ODESY-STATION).     000,0002819
017600 *     CHANGE FORMAT NAME                                             000,0002843
017700       MOVE "INVC" TO ODESY-FORMAT.                                   000,0002843
017800       MOVE 0 TO THIS-INV(ODESY-STATION).                             000,0002889
017900       GO TO APINVC-SUPPLY.                                           000,0002908
018000   APNAME-CHECK.                                                      000,0002925
018100       MOVE APNAME-NEXTVEND TO NEXT-VEND(ODESY-STATION).              000,0002925
018200       GO TO WRITE-REMOTE.                                            000,0002967
```


Figure 10.13 APINQ (5 of 7)

```
018300 /                                                                        000,0002984
018400   APINVC-SUPPLY.                                                         000,0002984
018500       MOVE THIS-INV(ODESY-STATION) TO INVOICE-KEY.                       000,0002984
018600   READ-INVOICE.                                                          000,0003003
018700       ADD 1 TO INVOICE-KEY.                                              000,0003003
018800       READ INVOICE-FILE INVALID KEY GO TO LAST-INVOICE.                  000,0003017
018900       IF THIS-VEND(ODESY-STATION) NOT = INVC-VENDNO                      000,0003289
019000 *        THEN READ NEXT INVOICE                                          000,0003289
019100           GO TO READ-INVOICE.                                            000,0003289
019200 *        ELSE SUPPLY INVOICE DETAILS                                     000,0003325
019300       MOVE INVOICE-KEY TO THIS-INV(ODESY-STATION).                       000,0003325
019400       MOVE SPACES TO ODESY-DATA.                                         000,0003344
019500       MOVE INVC-VENDNO TO APINVC-VENDNO.                                 000,0003358
019600       MOVE INVC-TRANNO TO APINVC-TRANNO.                                 000,0003395
019700       MOVE INVC-INVNO TO APINVC-INVNO.                                   000,0003455
019800       MOVE INVC-DATE TO APINVC-DATE.                                     000,0003515
019900       MOVE INVC-DUE-DATE TO APINVC-DUE-DATE.                             000,0003575
020000       MOVE INVC-AMOUNT TO APINVC-AMOUNT                                  000,0003635
020100       IF INVC-AMOUNT < 0 MOVE "-" TO APINVC-SAMOUNT.                     000,0003635
020200       MOVE INVC-PAID TO APINVC-PAID.                                     000,0003748
020300       IF INVC-PAID < 0 MOVE "-" TO APINVC-SPAID.                         000,0003785
020400       MOVE INVC-PAYABLE TO APINVC-PAYABLE.                               000,0003861
020500       IF INVC-PAYABLE < 0 MOVE "-" TO APINVC-SPAYABLE.                   000,0003898
020600       MOVE INVC-DESC TO APINVC-DESC.                                     000,0003974
020700       MOVE 88 TO ODESY-LENGTH.                                           000,0004034
020800       GO TO WRITE-REMOTE.                                                000,0004056
020900   LAST-INVOICE.                                                          000,0004073
021000       MOVE 0 TO THIS-INV(ODESY-STATION).                                 000,0004073
021100 *     CHANGE FORMAT NAME                                                 000,0004092
021200       MOVE "VEND" TO ODESY-FORMAT.                                       000,0004092
021300       GO TO WRITE-REMOTE.                                                000,0004138
021400   END-OF-JOB.                                                            000,0004155
```


Figure 10.14 APINQ (6 of 7)

```
C O D E    D I C T I O N A R Y
    BYTE LENGTH   CODEFILE RELATIVE DISK ADR
    ...........   .........................
000 00000520      000008
    00000520      TOTAL CODE

D A T A    D I C T I O N A R Y
    BYTE LENGTH   CODEFILE RELATIVE DISK ADR
    ...........   .........................
000 00000717      000003

S-M A C H I N E    P A R A M E T E R S    (S C R A T C H P A D)

LENB=8, SEGB=0, DISPB=11, CCPXB=5, COPB=23, D.E.F=430, BDISPB=13
BASE RELATIVE ADDRESSES:
  DATA-SEGMENT-0=708, COP-TABLE=41

P R O G R A M    P A R A M E T E R    B L O C K

FIRST-EXECUTABLE-INSTRUCTION=0,0
INTERPRETER-NAME=COBOL      /INTERP
STATIC-CORE=5736 BITS
DYNAMIC-CORE=0 BITS
DATA DICTIONARY STARTS AT CODEFILE SEGMENT 2, 1 ENTRY
CODE DICTIONARY STARTS AT CODEFILE SEGMENT 7, 1 ENTRY
FILE PARAMETER BLOCKS START AT CODEFILE SEGMENT 11, 3 ENTRIES

LAST ERROR AT SEQUENCE NUMBER       . 2 WARNINGS
***** COMPILATION COMPLETE
ELAPSED TIME : 02 MINUTES, 30 SECONDS
PROGRAM REQUIRES 14 DISK SEGMENTS OF 180 BYTES EACH.
MEMORY REQUIREMENTS
  0000520 BYTES = LARGEST CODE SEGMENT
  0000717 BYTES = BASE-TO-LIMIT AREA
  0000325 BYTES = DICTIONARIES AND RUN STRUCTURE
  0001132 BYTES = FILE BUFFERS & FILE INFO AREAS - INCLUDES 129 BYTES (+72 TO 540 IF DISK) FOR EACH FILE
  0002694 BYTES = ESTIMATED MEMORY REQUIRED TO RUN IF ALL FILES OPEN
215 SYMBOLIC RECORDS COMPILED AT 85.980 RECORDS PER MINUTE.
```


Figure 10.15 APINQ (7 of 7)

APPENDIX A

B1700 ODESY COMMANDS

/ACCEPT

/BYE [<operator number>]

/CLOSE

/DELETE <record number>

/HELLO <operator number> <password>

/INSERT <record number> <format>

/OPEN { [<batch number>] [ENTER] [<group>] }
{ <batch number> [VERIFY] [<group>] }
[INQUIRY]

/RELEASE [<batch number> [<batch number 2>]] [OVERRIDE / DELETE]

/STATUS [<batch number> / TOTALS] [ALL]

// { <format> / <record number> }

/ [{ ± } <increment> / <record number>]

/BREAK [<lsn>]

/BUG [ON / OFF]

/CONTROL <lsn>

/DS <program name>

/READY <lsn>

/TERMINATE [QC / NOW / <program name>]

/SS [<lsn> <text> / ALL <text>]

/WRU

N.B. commands entered from the console are not preceded by a slash ("/")

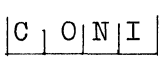
APPENDIX B.

COMMANDS FOR FORMAT MAINTENANCE

/AMEND	[GROUP FORMAT [FIELD]	<seq no>]
/BYE		
/DELETE	[GROUP FORMAT [FIELD]	<seq no>]
/DISPLAY		
/DUMP	<file name>	[<group name> [<format name>]]
/GET	<group name>	[<format name>]
/HELLO	<operator number>	<password>
/INSERT	[GROUP FORMAT [FIELD]	<seq no>]
/LIST	[GROUP FORMAT [FIELD]	<seq no>]
/LOAD	<file name>	[<group name> [<format name>]]
/OPTIONS	<terminal type>	- Specifies TD700/TD800 Braces/Brackets
/PD	[<group name>]	
/PRINT	[WORK AREA <group name>	[<format name>]]
/REMOVE	<group name>	[<format name>]
/SAVE		
/SS	<text>	
/TERMINATE		
/	- repeat, for use with INSERT, LIST, AMEND	
<data>		

Group Name SYS

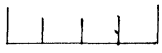
Format Name CONF



maintenance type (a constant), use INIT to create new files

maximum number of characters of constant data to be stored in the Edit, range: 100-4095 characters

maximum number of formats to be stored in the Edit, range: 1 - 63 formats



maximum number of fields to be stored in the Edit, range: 1 - 2047 fields

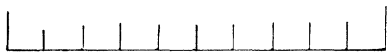
maximum number of user programs, to be used by the Edit, range 0-31 user programs.

maximum input size for any format, to be used by the Edit, range 1 - 1023, default = 160

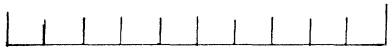
record size for the Tank file: a = 180 bytes, B = 540 bytes, C = 1080 bytes, default = A.

page size to be used for system reports, range: 10 - 99 lines

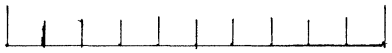
date type: U = MMDDYY, I = YYMMDD, E = DDMMYY, default = E.



name of pack on which the network controller resides



family name of network controller



file name of network controller

APPENDIX C.1

SYSTEM CONFIGURATION PARAMETERS

SHEET 1 OF 3

Group Name SYS

Format Name CONF

C O N I	maintenance type
0 2 0 0	allocate 200 characters for storing constants
0 6	6 formats to be stored in the Edit
0 1 0 0	100 fields to be stored in the Edit
0 1	one user program
0 1 6 0	default format input size
A	default Tank File record size
6 0	use 60 lines per page in system reports
E	default date type
	the network controller will
T E X T	reside on the system disk and
H A N D L E R	be called TEXT/HANDLER

EXAMPLE OF SYSTEM CONFIGURATION PARAMETERS

SHEET 1 OF 3

Group Name SYS Format Name CON2

C N 2 I	maintenance type (a constant)
	maximum number of characters of constant data to be stored in FORMAINT (constant characters per format) range: 100 - 4095
	maximum number of characters of scratchpad constant data to be stored in FORMAINT (constant characters per field) range: 25 - 1023
	maximum number of fields to be stored in FORMAINT (fields per format), range: 63 - 255
	maximum station buffer size for terminal with which FORMAINT is to be used. A = 300, B = 1024, C = 2000, D = 4000, default = C. (use B for display size of 960, C for display size of 1920 and D for display size of 3840)
	maximum number of formats to be stored in EXTRACT range: 1 - 31 formats
	maximum number of fields to be stored in EXTRACT range: 1 - 1023 fields
	maximum output record size (record size of an output file), to be used by EXTRACT. range: 80 - 999 bytes.

Group Name SYS Format Name CON2

|C|N|2|I|	maintenance type
|0|5|0|0|	allocate 500 characters for constant data in FORMAINT
|0|1|0|0|	allocate 100 characters for scratchpad constant data in FORMAINT
|0|6|3|	63 fields to be stored in FORMAINT
|C|	default station buffer size
|1|0|	10 formats to be stored in EXTRACT
|0|2|0|0|	200 fields to be stored in EXTRACT
|4|0|0|	400 byte output record to be used by EXTRACT

EXAMPLE OF SYSTEM CONFIGURATION PARAMETERS

SHEET 2 OF 3

Group Name SYS Format Name CON3

| C | N | 3 | I | maintenance type (a constant)

☐	duplicate anything (for duplicate in option)	☐	Include CDV
☐	restrict ACCEPT	☐	Include Batch Totals
☐	restrict CONTROL	☐	Include Crossfooting
☐	restrict DELETE	☐	Include ACCEPT
☐	restrict INSERT	☐	Include CONTROL
☐	restrict RELEASE	☐	Include DELETE
☐	restrict RELEASE DELETE	☐	Include DISPLAY
☐	restrict extended RELEASE	☐	Include INSERT
☐	restrict TERMINATE	☐	Include READY
		☐	Include extended RELEASE
		☐	Include SS
		☐	Include STATUS
		☐	Include WRU
		☐	Include DS

Possible values:

1. for duplicate anything
   Y = duplicate anything
   N = restricted to where previous record was entered with same format, default = N
2. for restrict option
   R = command restricted to supervisors
   N = command not restricted
   default = N
3. for include option
   I = include command or option
   E = exclude command
   default = I

Group Name SYS

Format Name CON3

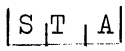
| C | N | 3 | I | maintenance type

Y	duplicate anything	
R	ACCEPT and CONTROL command restricted to supervisors only	
R		
N	DELETE, INSERT,	
N	RELEASE, extended	
N	RELEASE, RELEASE	
N	DELETE and TERMINATE	
N	commands not	
N	restricted	
		E CDV and Batch total
		E option excluded
		I Crossfooting option included
		I
		I
		I
		I
		I all commands
		I included
		I
		I
		I
		I
		I

EXAMPLE OF SYSTEM CONFIGURATION PARAMETERS

SHEET 3 OF 3

Group Name SYS Format Name STA



maintenance type (a constant)

maintenance action, I = insert, D = delete

logical station number (from NDL listing), range: 1 to 99

station name (from NDL listing)

station buffer size (from NDL listing), range 10 to 4095

forms set A = braces "{", B = brackets "[", C = US and RS, default = A

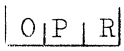
Control status, C = Control station, blank = not control station, default = blank

APPENDIX C.2. STATION DEFINITION PARAMETERS

S T A	maintenance type
I	insert new station
0 1	logical station number 1
T D 8 0 2 A	station name
1 9 2 0	1920 character buffer
	default forms definition characters, i.e. " {" and " }"
	not control station

EXAMPLE OF STATION DEFINITION PARAMETERS

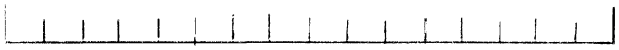
Group Name SYS Format Name OPR



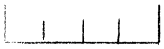
maintenance type (a constant)

maintenance action, I = insert,
D = delete

operator number, range: 1 to 99



operator name

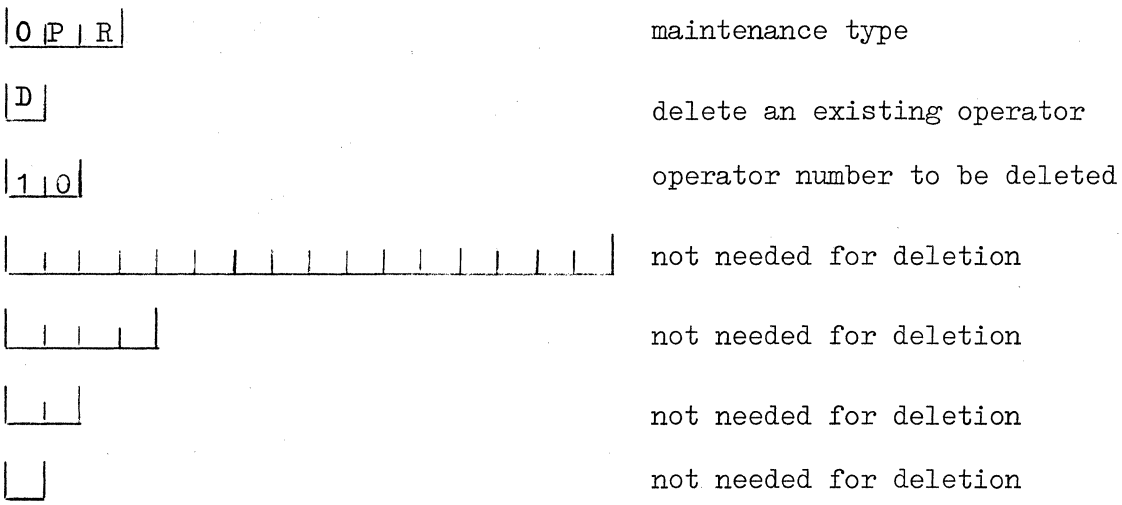


operator password

operator security level, range: 0 to 99

operator type, 0 = operator,
S = supervisor, default = 0

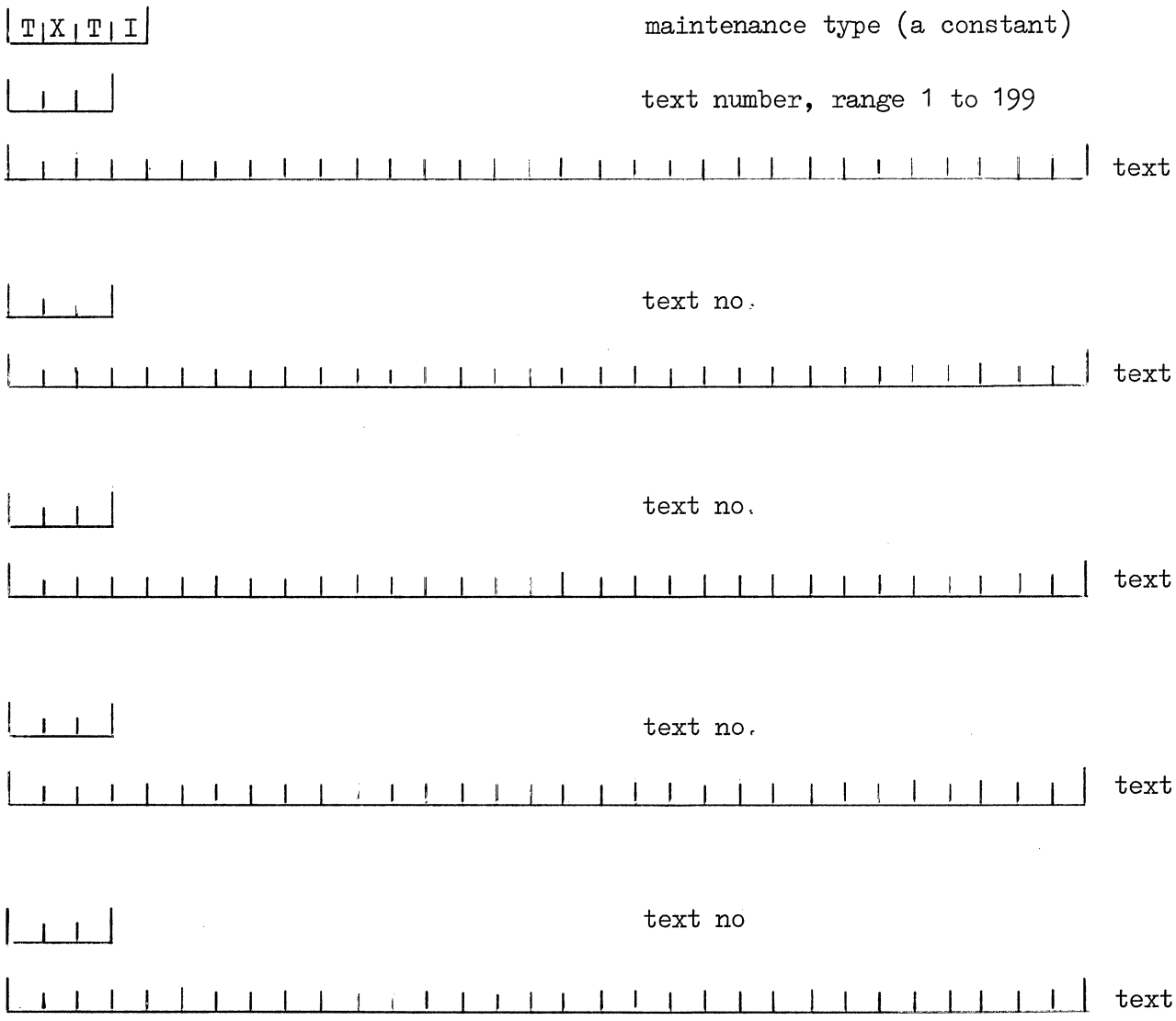
APPENDIX C.3 OPERATOR DEFINITION PARAMETERS



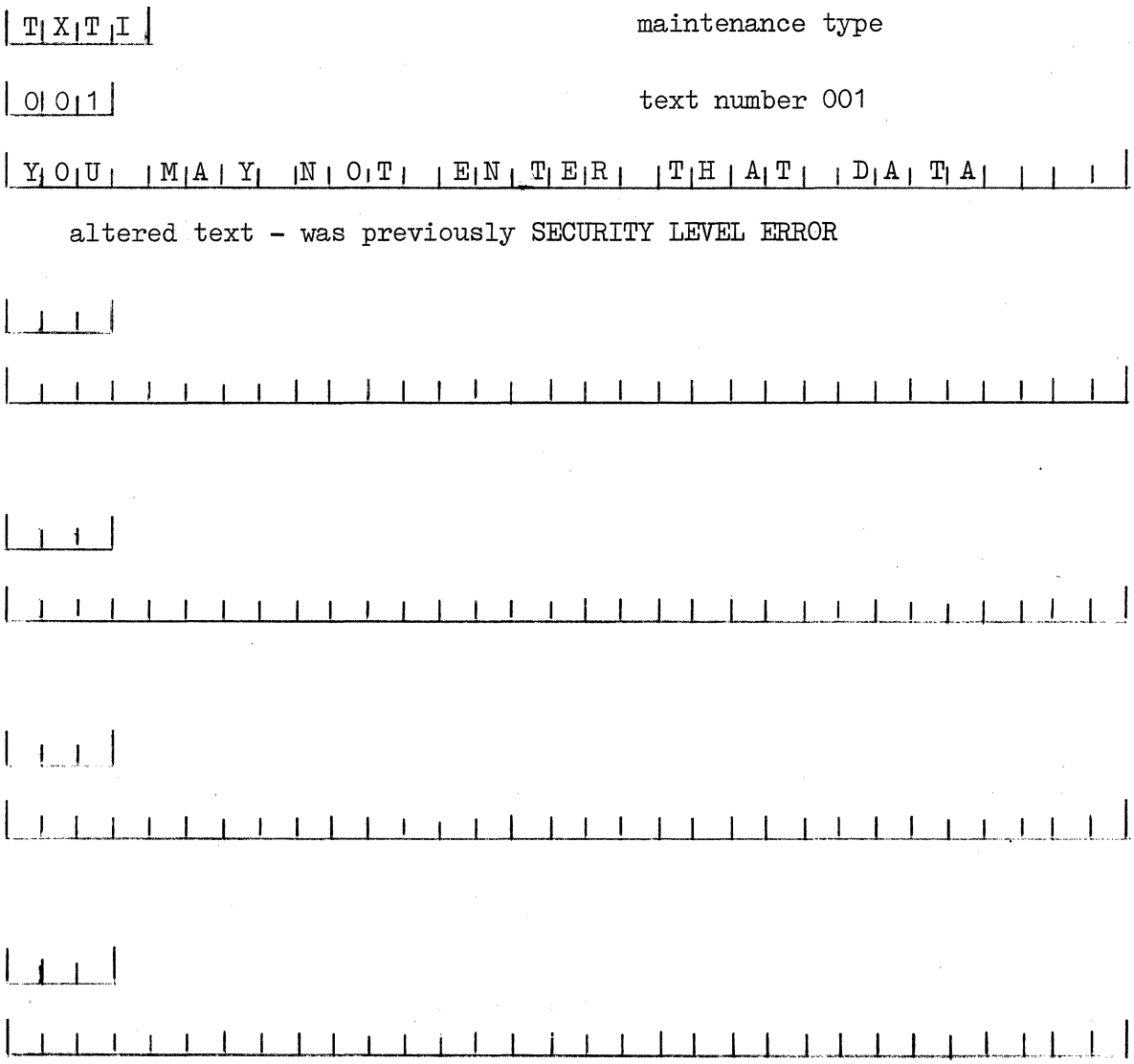
EXAMPLE OF OPERATOR DEFINITION PARAMETERS

Group Name SYS

Format Name TEXT



APPENDIX C.4 INFORMATION AND ERROR TEXT DEFINITION



EXAMPLE OF INFORMATION AND ERROR TEXT DEFINITION

Group Name SYS

Format Name CMD

| C | M | D | I | | 1 |

	CLOSE		ACCEPT
	DELETE		INSERT
	OPEN		RELEASE
	BYE		HELLO
	STATUS		BREAK
	SS		WRU
	DUMP		TERMINATE
	CONTROL		BUG
	READY		ENTER
	VERIFY		OVERRIDE

APPENDIX C5 EDIT COMMAND AND PARAMETER DEFINITION (1)

C	M	D	I		1

C	L	O
D	E	L
O	P	E
B	Y	E
S	T	A
S	S	
D	U	M
C	O	N
R	E	A
V	E	R

A	C	C
I	N	S
R	E	L
H	E	L
B	R	E
W	R	U
T	E	R
B	U	G
E	N	T
O	V	E

EXAMPLE OF EDIT COMMAND AND PARAMETER DEFINITION

Group Name SYS Format Name CMD2

| C | M | D | I | | 2 |

			ON		OFF
			NOW		QC
			ALL		DISPLAY
			MODIFY		TOTALS
			INQUIRY		DS

APPENDIX C6 EDIT COMMAND AND PARAMETER DEFINITION (2)

C	M	D	I		2

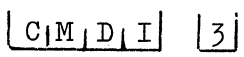
O	N		
N	O	W	
A	L	L	
M	O	D	
I	N	Q	

O	F	F	
Q	C		
D	I	S	
T	O	T	
D	S		

EXAMPLE OF EDIT COMMAND AND PARAMETER DEFINITION

Group Name SYS

Format Name CMD3



	INSERT		LIST
	AMEND		DELETE
	DISPLAY		SAVE
	GET		REMOVE
	PD		PRINT
	HELLO		BYE
	TERMINATE		DUMP
	LOAD		OPTIONS
	SS		GROUP
	FORMAT		FIELD

APPENDIX C7 FORMAINT COMMAND AND PARAMETER DEFINITION (3)

C	M	D	I		3

I N S	L I S
A M E	D E L
D I S	S A V
G E T	R E M
P D	P R I
H E L	B Y E
T E R	D U M
L O A	O P T
S S	G R O
F O R	F I E

EXAMPLE OF FORMAINT COMMAND AND PARAMETER DEFINITION

Group Name SYS Format Name CDV

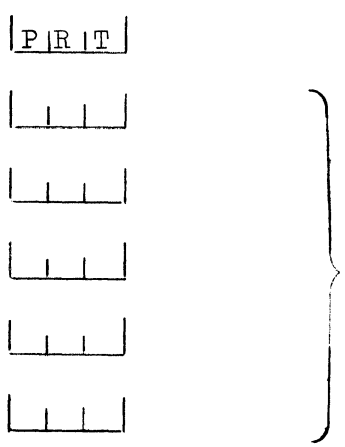
C D V	maintenance type (a constant)
	maintenance action, I = insert, D = delete
	scheme number, range: 1 to 9
	modulus, range 10 to 15
	remainder, range 1 to 15
	calculation type, E = products entered, A = products. calculated by addition, M = products calculated using the modulus, default = M
	complemented, Y = yes, N = no, default = N
	number of weights, range 1 to 10
	weights or products, enter hexadecimal equivalent.

APPENDIX C.8 CHECK DIGIT VERIFICATION SCHEME DEFINITION

C D V	maintenance type
I	insert a new scheme
1	scheme number 1
1 1	modulus 11
	remainder 0
	default product calculation type, i.e. modulus
	not complemented
0 6	6 weights
2 3 4 5 6 7	the weights

EXAMPLE OF CHECK DIGIT SCHEME DEFINITION

Group Name SYS Format Name PRNT

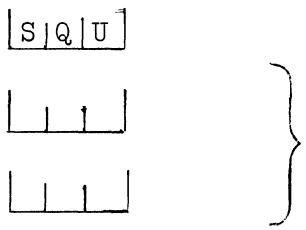


APPENDIX C.9 PRINT REQUEST DEFINITION

P R T	maintenance type
C T L	print control information on all files
C Ø N	print system configuration parameters

EXAMPLE OF PRINT REQUEST DEFINITION

Group Name SYS Format Name SQUA



maintenance type

squash request type, TNK = tank file,
FOR = formats file.

APPENDIX C.10 SQUASH REQUEST DEFINITION

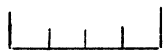
| S | Q | U | maintenance type

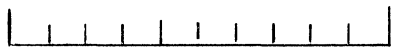
| T | N | K | squash tank file

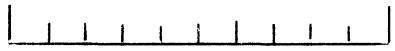
| F | O | R | squash formats file

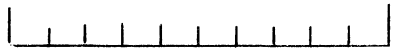
EXAMPLE OF SQUASH REQUEST DEFINITION

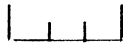
Action: /INSERT GROUP


group name

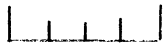

packname for output file


family name for output file


file name for output file


record size in characters, range: 1 to 999
default = 180

block size in records, range: 1 to 99
default = 1


records per area, range: 1 to 9999
default = 500

number of areas, range: 1 to 99
default = 40

report required, R = required
default = not required

medium, T = tape, default = any disk

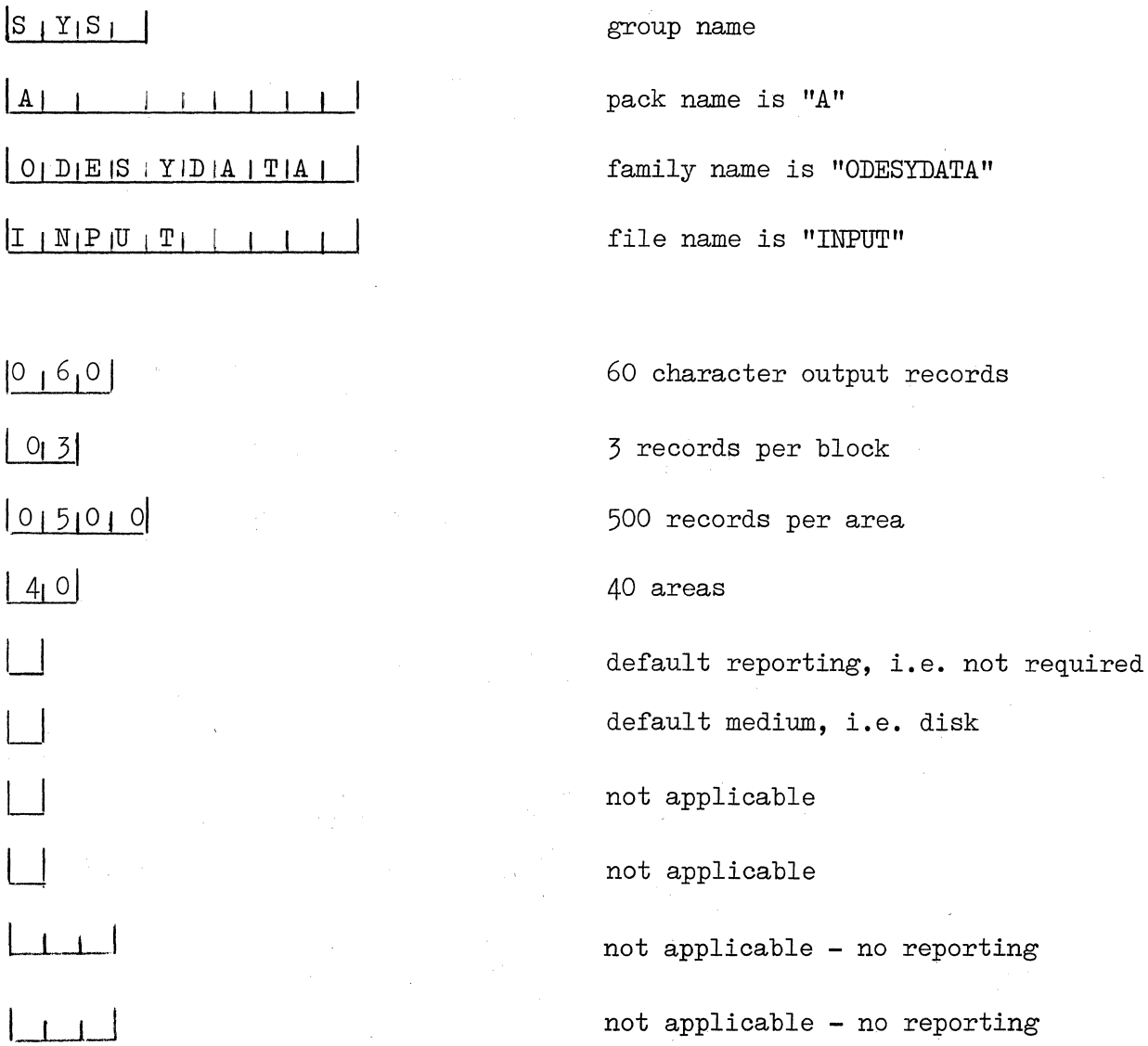
recording mode (tape only) A = ASCII, B = BCL,
default = EBCDIC (not yet implemented)

label type (tape only), A = ANSI, U = unlabelled,
default = Burroughs' standard

report input record number, blank = do not print
otherwise column number, range 1 to 127
default = blank

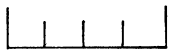
report output record number, blank = do not print
otherwise column number, range 1 to 127
default = blank

APPENDIX C.11 GROUP DEFINITION PARAMETERS (FORMAINT)

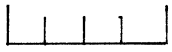


EXAMPLE OF GROUP DEFINITION PARAMETERS

Action: /INSERT FORMAT



format name



automatic following format

security level, range: 0 to 99

allow verification, N = verification not allowed, D = display verification default = verification allowed

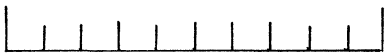
format type, blank = data entry format,

B = batch header format.

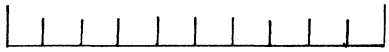
K = key data format, S = supply format,

C = check format, U = check and supply format

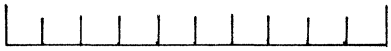
default = blank



pack name for user program

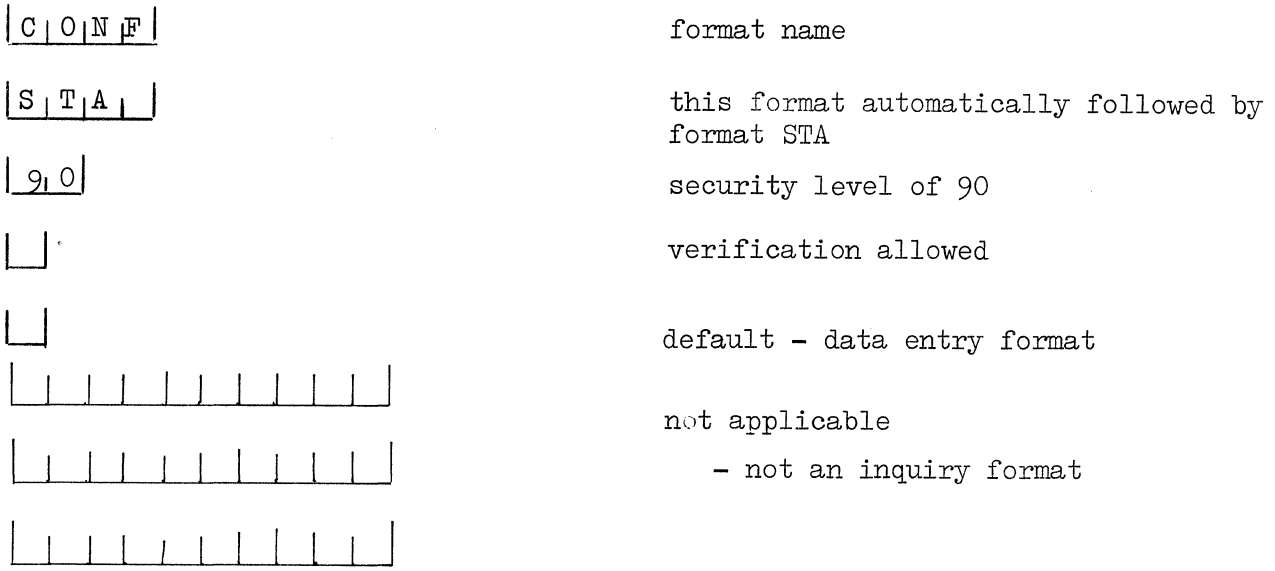


family name for user program



file name for user program

APPENDIX C.12 FORMAT DEFINITION PARAMETERS (FORMAINT)

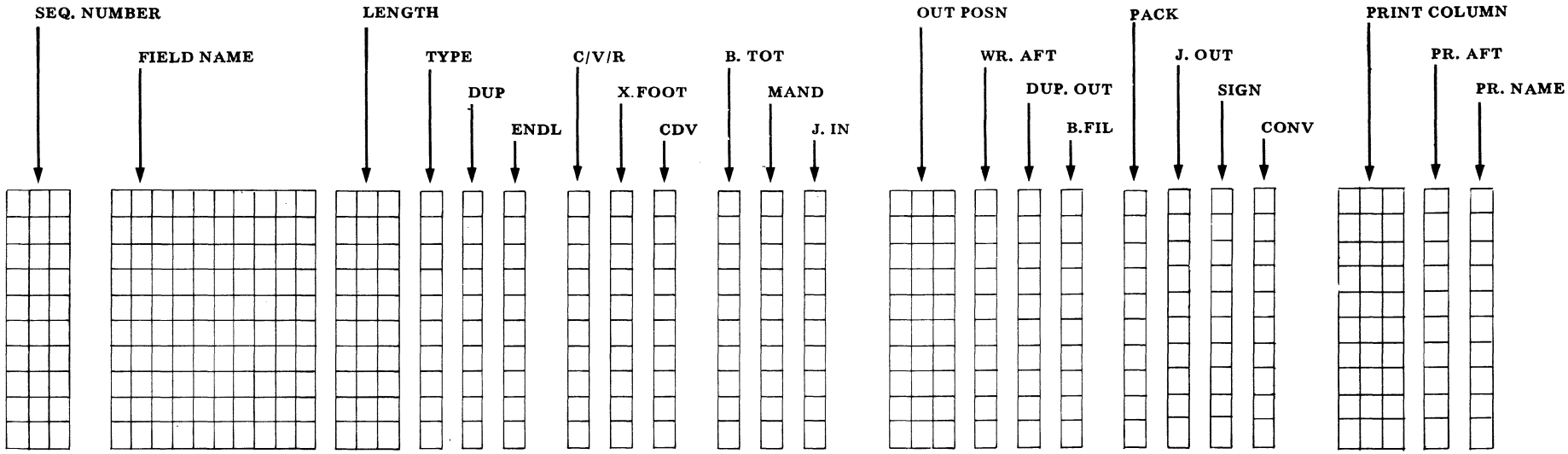


EXAMPLE OF FORMAT DEFINITION PARAMETERS

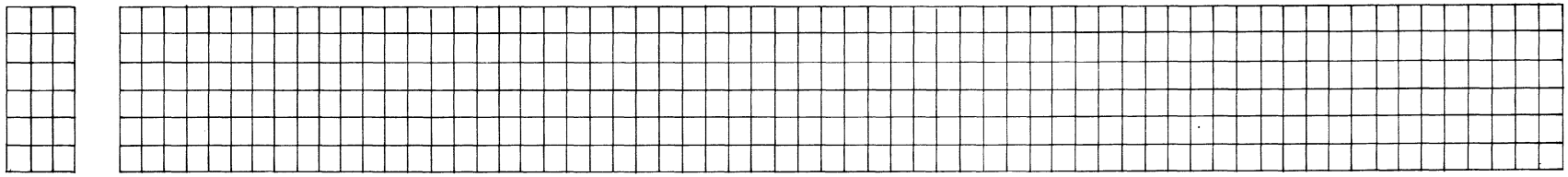
FIELD LAYOUT FORM

FORMAT

Page . . . . . of . . . . . .



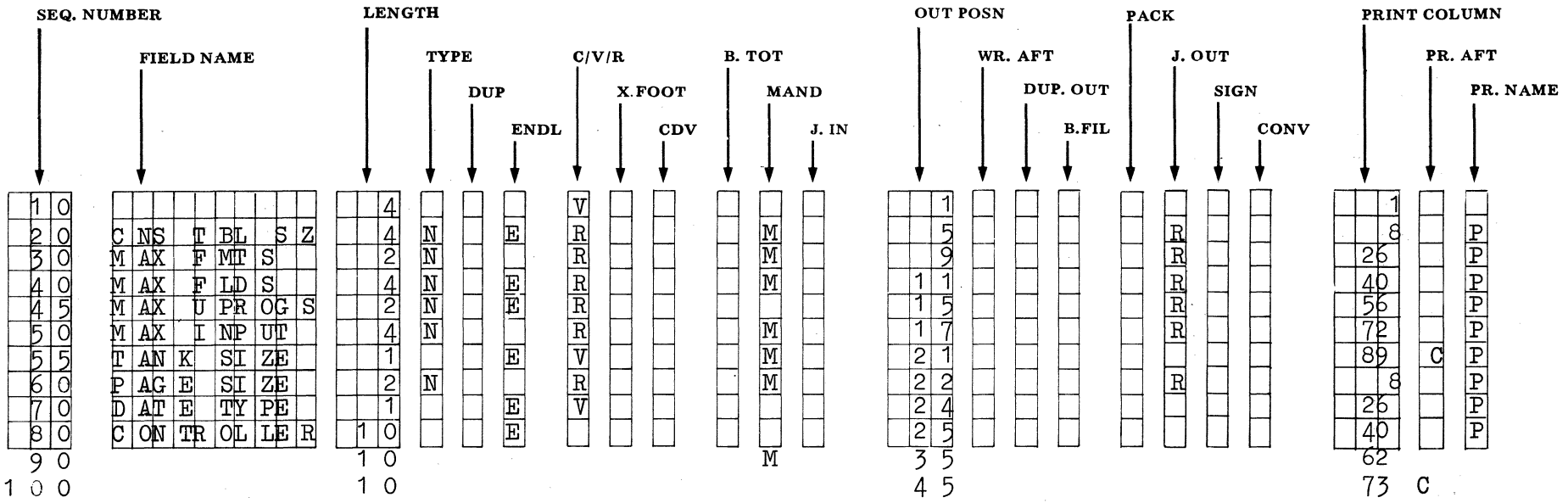
CONSTANTS



FIELD LAYOUT FORM

FORMAT SYS CONF

Page 1 of 1



CONSTANTS

```
10  CONI. INIT
20  0 20 0, 01 00 ,4 09 5
30  0 6, 01 ,6 3
40  0 10 0, 00 01 ,2 04 7
45  0 0  00  3 1
50  0 16 0 ,00 01 ,1 02 3
55  A , B, C
60  6 0, 10,99
70  E,U, I
```

```
B1700 DATA ENTRY SYSTEM      F O R M A T   M A I N T E N A N C E   R E P O R T      TIME 13:09  DATE 12/03/77  PAGE   1

   GROUP NAME  SYS                              FILE NAME            CDESYDATA /INPUT        DATE LAST CHANGED 15/09/76

   RECORD SIZE  80   BLOCKED  9   RECORDS/AREA  450   AREAS 40       ON DISK                              REPORT


   FORMAT NAME SYS CONF          NEXT FORMAT SYS CON2   SECURITY  50     D-VERIFY            DATE LAST CHANGED 09/02/77

SEQ    NAME   LGTH TYP DUP END CNS/VAL XFT CDV BTCT MAND JUST   CRT WRT DUP BLNK PACK JUST SIGN SIGN   PRINT PRINT PRINT
                    IN  LINE /RANGE                     IN    FES AFT OUT FILL       CLT      CONV   COL   AFTER NAME
 10             4                V                              1                                      1
                CONSTANTS   CONI INIT
 20 CNS TBL SZ  4  N       E     R                 M            5                R                     8         P
                CONSTANTS   0200 0100 4095
 30 MAX RMTS    2  N             R                 M            9                R                     26        P
                CONSTANTS   06 01 63
 40 MAX FLDS    4  N       E     R                 M           11                R                     40        P
                CONSTANTS   0100 0001 2047
 45 MAX UPROGS  2  N       E     R                             15                R                     56        P
                CONSTANTS   00 00 31
 50 MAX INPUT   4  N             R                 M           17                R                     72        P
                CONSTANTS   0160 0001 1023
 55 TANK SIZE   1           E     V                 M           21                                      89    C   P
                CONSTANTS   A B C
 60 PAGE SIZE   2  N             R                 M           22                R                     8         P
                CONSTANTS   60 10 99
 70 DATE TYPE   1           E     V                             24                                      26        P
                CONSTANTS   E L I
 80 CONTROLLER  10          E                                   25                                      40        P
 90             10                                M           35                                      62
100             10                                            85  C                                   73    C

**** TOTALS ****          FIELDS INPUT   12          FIELDS OUTPUT  12          FIELDS PRINTED   12
                          INPUT DATA SIZE   54       CHARACTERS OF CONSTANT DATA   68
```

D-1

Appendix D1 Format maintenance report for system configuration format

B1700 DATA ENTRY SYSTEM F I L E M A I N T E N A N C E R E P O R T TIME 11:21 DATE 12/03/77 PAGE 002

CONFIGURATION DATA

MAX STATIONS	CONSTANT TABLE	MAX FORMATS	MAX FIELDS	MAX USER PROGRAMS	MAX INPUT	TANK RECORD SIZE	PAGE SIZE
01	0200	06	0100	00	0160	A 180	60

NETWORK CONTROLLER NAME

TEXT /HANDLER

MAX CONST FMNT	MAX SCRATCHPAD FMNT	MAX FIELDS FMNT	MAX BUFFER FMNT	MAX FORMATS EXTR	MAX FIELDS EXTR	MAX OUTPUT EXT
0500	0100	063	C 2000	10	0200	400

RESTRICT OPTIONS

DUP.ANY	ACCEPT	CONTROL	DELETE	INSERT	RELEASE	REL.DEL	EXT.REL	TERMINATE
N	N	N	N	N	N	N	N	N

INCLUDE OPTIONS

CDV	BTCH.TCT	>FOOT	ACCEPT	CONTROL	DELETE	DISPLAY	INSERT	READY	EXT.REL
I	I	I	I	I	I	I	I	I	I

SS	STATUS	>RU	DS
I	I	I	I

Appendix D2 Example of a file maintenance report (for configuration data)

B1700 DATA ENTRY SYSTEM R E L E A S E R E P O R T TIME 12:08 DATE 12/03/77 PAGE 1

LAST REPORT: 3 DATED: 12/03/77 REPORT: 4

BATCH	RLSD BY	GROUP NAME	OUTPUT MEDIUM	(PACK)	FILE NAME	RECORDS IN BATCH ENTERED	VERIFIED	RE-EXTD	OUTPUT RECORDS BEFORE	AFTER	LINES PRINTED	
1	1	SYS				19	0					BATCH DELETED
28	1	MULT	DISK		MULTFILE	11	0	0	4	14	10	BATCH RELEASED
						11	0					BATCH DELETED
32	1	TEST	DISK		TESTFILE	6	6	6	40	46	10	BATCH RELEASED
33	1	SYS	DISK		ODESYDATA /INPUT	3	3	3	3	6	5	BATCH RELEASED

Appendix D3 Release report

REPORT: 1

OPERATOR REPORT

OPERATOR	NUMBER OF RECORDS ENTERED / VERIFIED	NUMBER OF KEYSTROKES ENTER MODE / VERIFY MODE	TOTAL ELAPSED TIME (HH:MM)	RECORDS PER HOUR (ENTER + VERIFY)	KEYSTROKES PER HOUR (ENTER + VERIFY)
1 SUPERVISOR 1	150 / 25	4285 / 907	5:46	30	900
2 SUPERVISOR 2	67 / 11	2533 / 328	3:34	21	802
3 SUPERVISOR 3	31 / 0	1010 / 0	1:07	27	904
4 SUPERVISOR 4	0 / 0	0 / 0	0:12	0	0
5 SUPERVISOR 5	0 / 0	0 / 0	0:00		
10 OPERATOR 10	0 / 0	0 / 0	0:00		
11 OPERATOR 11	0 / 0	0 / 0	0:00		
12 OPERATOR 12	0 / 0	0 / 0	0:00		
13 OPERATOR 13	0 / 0	0 / 0	0:00		
14 OPERATOR 14	0 / 0	0 / 0	0:00		
15 OPERATOR 15	0 / 0	0 / 0	0:00		
TOTALS	248 / 36	7828 / 1235	10:39	26	850

Appendix D4 Operator statistics report

REPORT: 1

FORMATS REPORT

GROUP SYS

FORMAT	RECORDS ENTERED
CONF	2
CON2	0
CON3	0
STA	3
CPR	27
TEXT	13
CMD	4
CMD2	4
CMD3	4
CDV	5
PRNT	39
TOTAL FOR GROUP	101

Appendix D5 Format statistics report for SYS group

BATCH 32 DISK FILE TESTFILE

BATCH TOT1 + 1
BATCH TOT2 2
BATCH TOT3 3
BATCH TOT4 + 4
BATCH TOT5 5

BT1 +00001	BT2 00001	BT3 0001	BT4 +01	BT5 01
BT1 +00000	BT2 00001	BT3 0001	BT4 +01	BT5 01
BT1 +00000	BT2 00000	BT3 0001	BT4 +01	BT5 01
BT1 +00000	BT2 00000	BT3 0000	BT4 +01	BT5 01
BT1 +00000	BT2 00000	BT3 0000	BT4 +00	BT5 01

B1700 DATA ENTRY SYSTEM B A T C H S T A T U S P R I N T R E P O R T TIME 12:07 DATE 12/03/77 PAGE 1

BATCH	NUMBER OF RECORDS	GROUP	OPEN OPR	REL CPR	STATUS FLAGS ENTER	VERIFY	EXTRACT	DELETE	DATE CREATED	DATE LAST ACCESSED
1	19	SYS	1	1	COMPLETED			DELETED	12/03/77	12/03/77
19	9	INQU	1	1	COMPLETED				03/03/77	12/03/77
22	7	ENGG	1	1	COMPLETED				10/03/77	12/03/77
25	6	INVC	1	1	COMPLETED		DONE		12/03/77	12/03/77
28	11	MULT	1	1	COMPLETED		SCHEDULED	DELETED	12/03/77	12/03/77
30	16	ENGG	1	1	COMPLETED				11/03/77	12/03/77
32	6	TEST	2	1	COMPLETED	COMPLETED	SCHEDULED		11/03/77	12/03/77
33	3	SYS	1	1	COMPLETED	COMPLETED	SCHEDULED		11/03/77	12/03/77
34	2	INVC	2	0	COMPLETED				11/03/77	11/03/77

Appendix D7 Batch status print report

APPENDIX E CHECK DIGIT VERIFICATION

E.1 GENERAL

ODESY allows for up to 9 check digit schemes to be defined in the System file. This appendix explains the basic principles employed in check digit calculation and gives examples of the schemes most commonly used.

Each scheme has a number (1 - 9 in the case of ODESY) and a modulus (10 - 15). There are also weights, which are the amounts by which each digit of the integer is multiplied to arrive at a product, and possibly a remainder which is added to the sum of the products. The sum of the products is taken away from the next highest multiple of the modulus either using a remainder or not.

E.2 EXAMPLES

To calculate the check digit for an integer

5 6 8 3 9 6

using modulus 10 and weights 2, 1, the following calculation would be performed:

integer	5 6 8 3 9 6
weights	1 2 1 2 1 2
products	5 12 8 6 9 12

The products may then be added

$5 + 12 + 8 + 6 + 9 + 12 = 52$

or the individual digits in the products may be added : $5 + 1 + 2 + 8 + 6 + 9 + 1 + 2 = 34$

The sum of the products is then subtracted from the next highest multiple of 10.

In the first case the check digit is 8 and in the second case 6. If a remainder is used this is added to the sum and the sum plus remainder is then subtracted from a multiple of the modulus. If the check digit is to be complemented then the result is subtracted from the modulus, giving 2 and 4 respectively in the above examples.

Some commonly used check digit schemes are as follows:-

1. Burroughs modulus 10 and IBM modulus 10 method

Modulus 10

Remainder 0

Weights 2,1,2,1,2,1 .....

Each individual digit of the product is added

Example: integer 8 3 9 6 5 6

weights 1 2 1 2 1 2

sum of product 8+6+9+1+2+5+1+2+ = 34

40 - 34 = 6

number and check digit 8396566

2. IBM modulus 11 method

Modulus 11

Remainder 0

Weights 2,3,4,5,6,7,2,3 etc .....

Example:

Integer 5 6 8 3 9 6

Weights 7 6 5 4 3 2

Sum of products 35+36+40+12+27+12 = 162

165 - 162 = 3

number and check digit 5683963

3. Burroughs modulus 11 method

Modulus 11

Remainder 1

Weights 7,3,1,7,3,1

Example:

Integer	5	6	8	3	9	6	
Weights	1	3	7	1	3	7	
Sum of products	5+18+56+3+27+42					=	151 + remainder (1)

154 - 152 = 2

number + check digit = 5683962

The Edit uses tables of products and weights which it references for check digit calculation. Each weight is represented by a set of ten products corresponding to the digits 0 - 9. In the Burroughs modulus 10 scheme, which uses the weights 2 and 1, the table is as follows:

WEIGHT	ZERO	ONE	TWO	THREE	FOUR	FIVE	SIX	SEVEN	EIGHT	NINE
2	0	2	4	6	8	1	3	5	7	9
1	0	1	2	3	4	5	6	7	8	9

Here the individual digits are added to give a single figure product

e.g. six x 2 = 12, 1 + 2 = 3

Using the example 839656 and starting with the least significant digit with the first weight

6 (first weight) gives 3

5 (second weight) gives 5

6 (first weight) gives 3

9 (second weight) gives 9

3 (first weight) gives 6

8 (second weight) gives 8

34

40 - 34 = 6

Number and check digit = 8396566

Check digit schemes may be defined on the Filemaint input form in one of two ways. The user may either specify a calculation type of E and enter the products, or they may specify a calculation type of M or A and enter the weights. In the case of M or A the weights will be used to calculate the products. The following forms illustrate two alternative ways of specifying the Burroughs modulus 10 scheme.

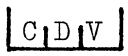
Group Name SYS Format Name CDV

C D V	maintenance type (a constant)
I	maintenance action, I = insert, D = delete
1	scheme number, range: 1 to 9
1 0	modulus, range 10 to 15
	remainder, range 1 to 15
E	calculation type, E = products entered, A = products calculated by addition, M = products calculated using the modulus, default = M
N	complemented, Y = yes, N = no, default = N
0 2	number of weights, range 1 to 10
0 2 4 6 8 1 3 5 7 9	
0 1 2 3 4 5 6 7 8 9	weights or products, enter hexadecimal equivalent

EXAMPLE 1 - ENTERED PRODUCTS

Group Name SYS

Format Name CDV



maintenance type (a constant)

maintenance action, I = insert, D = delete

scheme number, range: 1 to 9

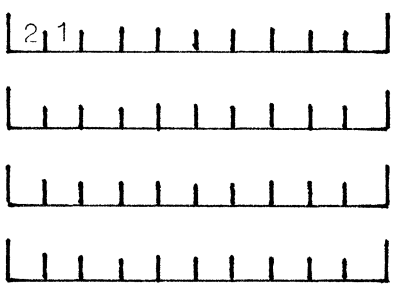
modulus, range 10 to 15

remainder, range 1 to 15

calculation type, E = products entered,
A = products calculated by addition,
M = products calculated using the modulus,
default = M

complemented, Y = yes, N = no, default = N

number of weights, range 1 to 10



weights or products, enter hexadecimal equivalent

EXAMPLE 2 - PRODUCTS CALCULATED FROM THE WEIGHTS BY ADDITION

APPENDIX F ERROR AND INFORMATION MESSAGES AND COMMANDS

F.1 General

This appendix contains a list of error and information messages output by the Edit and FORMAINT, and a list of commands and parameters used by the same two programs.

Sections F2 and F3 contain the list of error messages. In each case the message number is listed, followed by the standard text, the maximum number of characters which may be altered by the user, the cause of the message and, where applicable, any action necessary to rectify the error. The messages output by the Edit are numbered 1 - 100, whereas those for FORMAINT are numbered 101 - 199.

Sections F4 and F5 contain the list of commands. In each case the command or parameter is listed, followed by the standard abbreviation for it.

F.2 MESSAGES USED BY THE EDIT

Msg No.	Text	Length	Cause	Action
1	SECURITY LEVEL ERROR	31	The operator security level is below the format security level.	Check that the security levels are correct.
2	CLOSING DOWN-END YOUR WORK NOW	31	A warning that the system will be shutting down. This message is sent when requested by a supervisor and causes an indicator to be set which will not allow any further operators to log in.	Operators should finish the batch that they are processing and then log out.
3	CLOSE FORCED - YOU ARE LOGGED OUT	31	The EDIT has terminated and all operators still logged in have been automatically logged out.	If not already logged out, the operator should note how many records have been processed in this batch.
4	SYSTEM ERROR - LAST COMMAND LOST	31	The last command keyed in cannot be executed because of an internal processing fault.	Rekey the command.
5	THE BATCH IS NOW CLOSED	31	The /CLOSE command has been executed successfully.	Continue with next batch.
6	THE RECORD IS ACCEPTED	31	Either the /ACCEPT command has been used or the data entered has been accepted by the EDIT as correct.	Continue with the next record.

Msg No.	Text	Length	Cause	Action
7	NOW SCHEDULED FOR RELEASE	31	The /RELEASE command has been used and the batch is about to be released.	The supervisor may continue with the next batch.
8	RECOVERY STARTED	31	It has been necessary to execute the recovery system before the program can be restarted. This message is displayed at the console.	None required.
9	A BATCH IS ALREADY OPEN	31	The /OPEN command has been used when the station at which this message appears already has a batch open. This message can appear if a supervisor has not closed a batch after using the /RELEASE command and then attempts to open another batch for release.	Finish processing the batch which is already open. Close this batch and then open the batch requested.
10	YOU ARE NOT ENTERING DATA	31	An attempt is being made to transmit data without having a batch opened in ENTER mode. This could happen if an operator had just keyed in /OPEN to display some records and then attempted to display a format for data entry. This message is also transmitted when an attempt is made to use /ACCEPT in the VERIFY mode.	Key in /CLOSE and then /OPEN ENTER.

Msg. No.	Text	Length	Cause	Action
11	AN OPERATOR IS LOGGED IN	31	A /HELLO command is being used at a station at which an operator is already logged in.	Either log in at another station or wait until the other operator has logged out at this station.
12	NO OPERATOR IS LOGGED IN	31	An attempt is being made to use a station without first having logged in.	Key in /HELLO <operator number> <password >and proceed as normal.
13	YOU ARE LOGGED IN ELSEWHERE	31	An operator is attempting to log in at one station without having logged out at another station.	Key in /BYE at the station at which the operator has not yet logged out and then /HELLO at the station at which the message appears. Alternatively, key in /BYE <operator number> at the station at which the message appears, followed by /HELLO <operator number> <password >.
14	YOUR NAME ISN'T ON FILE	31	A password or an operator number which is not defined in the System file is being used.	Check that the password is correct and that the number is exactly as it appears in the operator listing. Note that for reasons of security no passwords appear in the operator listing.

Msg No.	Text	Length	Cause	Action
15	I CAN'T LOAD THAT FORMAT	31	The format is too large to be stored in the EDIT's internal tables.	The person responsible for setting up this format should be consulted to see what can be changed to accommodate it.
16	THAT FORMAT DOESN'T EXIST	31	The format requested is not stored in the Formats file.	Check that the correct format name has been requested.
17	THAT BATCH IS IN USE ELSEWHERE	31	An /OPEN command has been used to specify a batch that is already open at another station.	Wait until the batch has been closed at the other station and then rekey the /OPEN command.
18	NO BATCH IS OPEN	31	An attempt is being made to manipulate data without having opened a batch.	Key in /OPEN <batch no >.
19	THE BATCH WILL BE DELETED	31	A supervisor is satisfied that the batch has undergone all the necessary processing and has used the /RELEASE DELETE command. The batch will be deleted when the extraction program next runs.	None required.
20	THAT COMMAND IS RESTRICTED	31	An operator is attempting to use a command which has been restricted to supervisors only.	The operator should consult his or her supervisor.

Msg No.	Text	Length	Cause	Action
21	THE RECORD IS DELETED	31	The /DELETE command has been used to successfully delete a record.	None required.
22	THE RECORD IS INSERTED	31	The /INSERT command has been used to successfully insert a record.	None required.
23	THE RECORD DOES NOT EXIST	31	A record which is not in the batch being processed at this station has been requested. The record may either be in another batch or it may not exist at all.	Check that the record number has been keyed in correctly. Check whether the record has been deleted.
24	YOU ARE LOGGED IN	31	A /HELLO command has been successfully keyed in.	Continue processing.
25	YOU ARE LOGGED OUT	31	A /BYE command has been successfully keyed in.	None required.
26	I AM NOT EXPECTING DATA	31	An operator is attempting to key in data when there is no format displayed.	Use // <format > to display the format required.
27	ODESY IS RESTARTED	31	Recovery has been completed and ODESY is running again. (This message appears at the console only) See also message no. 8.	None required.

Msg No.	Text	Length	Cause	Action
28	LAST BATCH AND RECORD	21	Recovery took place on restarting ODESY. This message gives the last batch and record processed at this station.	None required.
29	BATCH OPENED	31	The /OPEN command has been used to successfully open a batch.	Continue processing.
30	I DON'T RECOGNIZE THAT COMMAND	31	A command has been keyed in either incorrectly or spelt wrongly.	Key in the command exactly as shown in Appendix A.
31	INVALID STATION NUMBER	31	Either the lsn specified in a /BREAK, /READY or /CONTROL command was zero or greater than max stations, or the lsn specified in an /SS command was greater than max stations.	Check whether the number for max stations in the system file has been altered and also check the number of the station required.
32	WARNING SENT TO STATIONS	31	The TERMINATE command has been used at the console. This message will also appear at the console.	None required.
33	THE SYSTEM IS CLOSING DOWN	31	An operator is attempting to log into the system after the shut-down cycle has been initiated.	Wait until the system is run again.

Msg No.	Text	Length	Cause	Action
34	MADE CONTROL STATION	31	A /CONTROL command has been used successfully to specify that a certain station will be able to perform control functions.	None required.
35	INVALID SYNTAX	31	The parameters specified with a certain command are not correct.	Re-key the correct parameters for this command.
36	THAT GROUP DOES NOT EXIST	31	An /OPEN command is being used and specifies a group name that is not in the Formats file.	Check whether the group name has been keyed in correctly and whether this group has been deleted from the Formats file.
37	THERE ARE NO FREE BATCHES	31	An /OPEN command has been used and no batch number has been specified. However, the edit is unable to allocate a batch number since all batches have been allocated.	Delete those batches which are scheduled for deletion and run the extraction program. The batch numbers may then be reallocated.
38	THE DESTINATION IS NOT ON-LINE	31	An /SS command has been used but the receiving station is not yet ready.	Try the command again after waiting a few moments.

Msg No.	Text	Length	Cause	Action
39	BATCH TOTAL COMPARISON ERROR	31	A /RELEASE command has been used but the batch totals entered by the operator do not match those calculated by the computer.	Either correct the operator-entered batch totals in the batch header record or use the OVERRIDE option to force release of the batch.

Message numbers 40 to 46 inclusive appear when the /STATUS command is used for a batch. Only a certain number of characters in each message may be altered by the user.

40	BATCH NUMBER AND RECS	21	Use of the /STATUS command causes this message to be sent as the first line of a batch status report. It gives the batch number and the total records in this batch. Only the first 21 characters of this message may be altered.	None required.
41	IN USE AT STATION	29	This gives the second line of the batch status report. It shows the logical station number of the station currently using the batch (if it is being used). Only the first 29 characters of the message may be altered.	

Msg No.	Text	Length	Cause	Action
42	ENTRY STATUS	21	This gives the first 21 characters of the third line of the batch status report. Message no. 58 forms the second half of the line, this being either IN PROGR or COMPLETE.	
43	VERIFICATION STATUS	21	This gives the first 21 characters of the fourth line of the batch status report. The second part of the line will be either blank, IN PROGR or COMPLETE (see message 58).	
44	DELETE STATUS	21	This gives the first 21 characters of the fifth line of the batch status report. The second part of the line will be either blank or DELETED. Message number 58 forms the second part of the line.	
45	EXTRACTION STATUS	21	This gives the first 21 characters of the sixth line of the batch status report. The second part of the line will be either blank, SCHEDULED or DONE. Message number 59 forms the second part of the line.	

Msg No.	Text	Length	Cause	Action
46	OPENED AND RELEASED BY	27	This gives the first 27 characters of the seventh line of the batch status report. The line is completed by the number of the operator or supervisor who released the batch.	

Message numbers 47 to 51 inclusive appear when the /STATUS command has been used for a station. Only a given number of characters in each message may be altered by the user.

47	STATION NUMBER AND OPERATOR	26	This gives the first 26 characters of the first line of a station status report. The second part of the line will be the logical station number of the station and the number of the operator, if any, logged in at this station.	
48	CURRENT & LAST BATCH	24	This gives the first 24 characters of the second line of the station status report.	
49	CURRENT RECORD & MODE	22	This gives the first 22 characters of the third line of the station status report. The second part of the line is formed by the number of the record currently	

Msg No.	Text	Length	Cause	Action
49	cont'd		being processed and the mode for the batch in use (ENTER,VERIFY).	
50	CURRENT GROUP & FORMAT	23	This gives the first 23 characters of the fourth line of a station status report. The second part of the line is formed by the names of the group and format currently in use at this station.	
51	LOGGED IN SINCE	26	This gives the first 26 characters of the fifth line of a station status report. The second part of the line is formed by the time at which an operator logged in at this station. (24 hour clock).	
52	THE BATCH DOES NOT EXIST	31	Either an /OPEN or a /STATUS command has been used to specify a batch number that is not valid.	Check that the correct batch number has been entered on the input form and that it has been keyed in correctly.
53	DONE	31	An /SS or /READY command has been performed successfully.	None required.

Msg. No.	Text	Length	Cause	Action
54	I WAS LOOKING FOR A NUMBER	31	A command which should have been followed by a number has been used followed by a name or by blanks.	Check the syntax of the command and rekey it correctly if necessary.
55	SHUTDOWN FORCED. TANK FILE FULL		There is no more room in the Tank file to store data.	Make more space in the Tank file either by deleting batches and then running the extraction program or by merely running the extraction program if batches are known to be scheduled for deletion.
56	YOU ARE IN DISPLAY MODE	31	A verify, modify, insert or delete command is being used when a batch is in display mode.	Use /OPEN VERIFY, or /OPEN ENTER and then rekey the relevant command.
57	BATCH CLOSED - NOT ALL VERIFIED	31	The /CLOSE command has been used but not all records have been verified in the VERIFY mode. This is only a warning.	None required.
58	IN PROGR COMPLETED DELETED	10	Used to form the second part of the line in message nos 42, 43 or 44, signifying that a particular mode is either in progress or completed, or that a batch is scheduled for deletion.	None required.

Msg No.	Text	Length	Cause	Action
59	SCHEDULED/DONE	10	Used to form the second part of the line in message 45 signifying that extraction is either scheduled or has been completed.	None required.
60	CONTROLLER MISSING	31	The EDIT is not able to find the network controller.	Check that the system configuration has been correctly entered and includes a network controller entry.
61	NOT ENOUGH RECOVERY SPACE	31	The EDIT has found that the Tank file recovery record is not large enough to hold all the information necessary for the requested number of stations.	Either reduce the number of stations or change the value of the Record Size option for the Tank File.
62	PLEASE RE-EXECUTE	31	The EDIT either has too much dynamic memory or not enough and has terminated.	Re-execute the EDIT.
63	EOF ON REMOTE FILE	31	The network controller may have been DSed while the EDIT is running.	Restart the EDIT.

Msg No.	Text	Length	Cause	Action
64	YOU CANNOT CHANGE GROUP	31	An attempt has been made to OPEN an existing batch and the group name specified was different to the original group name.	Check that the correct batch number has been specified
65	INQUIRY PROGRAM IS NOT RUNNING	31	A user-program, required by an inquiry format is not running	Repeat the command. If error occurs again, report to supervisor.
66	INVALID INQUIRY KEY	31	An invalid key, or no key, has been specified when using an inquiry format	Report to supervisor
67	INVALID INQUIRY FORMAT	31	The user program specified cannot handle the format	Report to supervisor
68	INVALID INQUIRY REPLY	31	The user program gave an invalid reply	Report to supervisor
69	FIRST UNVERIFIED RECORD	25	The batch has been closed in verify mode, but not all records have been verified. Message 57 is also sent.	None required
70	YOU ARE IN INQUIRY MODE	31	Either an /OPEN INQUIRY command has been received, or an invalid command for inquiry mode (DISPLAY, VERIFY, INSERT, DELETE) has been received.	If OPEN, then none required. If invalid command, then report to supervisor.

Msg No.	Text	Length	Cause	Action
71	NOT BUSY	31	Following /STATUS ALL or /STATUS <batch number>ALL, this message is sent when there are no more status reports to send.	None required
72	SYSTEM & TANK FILES DONT MATCH	31	The last access date/time of the 2 files is different Message 75 is also sent.	Load the most recent backup copies of the 2 files.
73	LINKAGE ERRORS IN TANK FILE	31	The tank file has been corrupted. Messages 74 and 75 are also sent	Execute the DUMP program, then load the most recent backup copies of the files.
74	PLEASE EXECUTE ODESY/DUMP THEN	31	See above	
75	PLEASE LOAD BACKUP FILES	31	See messages 72 and 73 above	
76	INQUIRY PROGRAM TABLE FULL	31	An inquiry format has been requested, but no room for new user program in table	Report to supervisor

F.3 MESSAGES USED BY FORMAINT

Msg No.	Text	Length	Cause
101	GROUP	7	This is used in the first line of control information for the Group Templates.
102	FORMAT	7	This is used in the first line of control information for the Format Template.
103	FIELD	7	This is used in the first line of control information for the Field Templates.
104	I-CONS	7	This is used in the first line of control information for the Inserted Constant Template.
105	R-CONS	7	This is used in the first line of control information for the Range Constants Template.
106	V-CONS	7	This is used in the first line of control information for the Value Constants Template.
107	INSERT	7	This is used in the first line of control information during Insertion of format data.
108	LIST	7	This is used in the first line of control information during Listing of format data.
109	AMEND	7	This is used in the first line of control information during Amendment of format data.
110	DISPLAY	7	This is used in the first line of control information during the Display of a format.

Msg No.	Text	Length	Cause
111	REMOVE	7	This gives the first seven characters of message 130 when the command was Remove.
112	DUMP	7	This gives the first seven characters of message 130 when the command was Dump.
113	LOAD	7	This gives the first seven characters of message 130 when the command was Load.
114	PRINT	7	This gives the first seven characters of message 130 when the command was Print.
115	OPTIONS	7	This gives the first seven characters of the reply to the Options command.
116	WORKAREA	8	This is used as a KEYWORD in the /PRINT WORK AREA command, and also gives the first eight characters of message 127 in the reply to that command.
117	ALL	8	This gives the first eight characters of messages 125 or 126 or 127, where no group or format was specified for Dump, Load or Print.
118	INSERTED	15	This gives the second half of the reply to the insertion of a field. The first half of the message will be message 103.
119	AMENDED	15	This gives the second half of the reply to the amendment of a group, format or field. The first half of the message will be message 101 or 102 or 103.

Msg No.	Text	Length	Cause
120	DELETED	15	This gives the second half of the reply to the deletion of a group, format or field. The first half of the message will be message 101, or 102 or 103.
121	NOT PRESENT	15	This gives the second half of the reply to List, Amend or Delete when the group, format or field is not in the Workarea. The first half of the message will be message 101, or 102 or 103.
122	DUPLICATE FIELD	15	This gives the second half of the reply to /INSERT FIELD <seq no> where the field sequence number entered already exists in the Workarea. The first half of the message will be message 103.
123	SAVED	15	This gives the second half of the reply to the Save command. The first half of the message will be the group name and format name.
124	REMOVED	15	This gives the second of the reply to the Remove command. The first half of the message will be the group name and format name.
125	DUMPED	15	This gives the second half of the reply to the Dump command. The first half of the message will be the group name and format name.

Msg No.	Text	Length	Cause
126	LOADED	15	This gives the second half of the reply to the Load command. The first half of the message will be the group name and format name.
127	PRINTED	15	This gives the second half of the reply to the Print command. The first half of the message will be the group name and format name.
128	NOW AVAILABLE	15	This gives the second half of the reply to the Get command. The first half of the message will be the group name and format name.
129	NOT IN FILE	15	This gives the second half of the reply to the Get, Print, Dump or Remove commands where the group or format names cannot be found in the Format file. The first half of the message will be the group name and format name.
130	STARTED	15	This gives the second half of the initial response to a Remove, Dump, Load or Print command. The first half of the message will be message 111, 112, 113 or 114.
131	NOT RECOGNISED	30	The last command was entered incorrectly.
132	NOT BUSY	30	The last input was data rather than a command, but the operator is not inserting, listing or amending at the moment.

Msg No.	Text	Length	Cause
133	SAVE OR DELETE	30	The operator is attempting to Insert, or Bye or Terminate without Saving or Deleting the Workarea.
134	MESSAGE ZIPPED TO MCP	30	This is the reply to any input preceded by a question mark.
135	START OF SESSION	30	This is the reply to a Hello command with a valid operator number and password.
136	END OF SESSION	30	This is the reply to a Bye or Terminate command.
137	ENTER OPERATOR/ PASSWORD	30	No operator is logged in
138	INVALID OPERATOR/ PASSWORD	30	This is the reply to a Hello command with an invalid operator number or password.
139	ALREADY LOGGED IN	30	This is the reply to a Hello command where the operator is already logged in.
140	TOO MANY FIELDS FOR FORMAT	30	This is the reply to an /INSERT FIELD <seq no> command where the number of fields in the Workarea is already the maximum allowed (normally 63 fields).
141	GROUP INDEX IS FULL	30	This is the reply to a Save or Load command where there are already 196 groups in the Format file.
142	CONSTANT TABLE OVERFLOW	30	This is the reply to the entry of format data where there are too many constants for the Workarea.

Msg No.	Text	Length	Cause
143	NO FORMATS HAVE BEEN LOADED	30	This is the reply to a load command where the specified format or formats were not found in the dump file.
144	FILE IS INCOMPATIBLE	30	This is the reply to a Load or Dump command where the specified dump file does not contain records of length 60 bytes, blocked 3.
145	FILE IS ALREADY IN USE	30	This is the reply to a Load or Dump command where the specified dump file is in use by another program.
146	FILE NOT ON DISK	30	This is the reply to a Load command where the specified dump file is not on disk.
147	MESSAGE SENT TO CONSOLE	30	This is the reply to the SS command

F.4 COMMANDS AND ABBREVIATIONS USED BY THE EDIT

	Command or Parameter	Abbreviation
1	CLOSE	CLO
2	ACCEPT	ACC
3	DELETE	DEL
4	INSERT	INS
5	OPEN	OPE
6	RELEASE	REL
7	BYE	BYE
8	HELLO	HEL
9	STATUS	STA
10	BREAK	BRE
11	SS	SS
12	WRU	WRU
13	DUMP	DUM
14	TERMINATE	TER
15	CONTROL	CON
16	BUG	BUG
17	READY	REA
18	ENTER	ENT
19	VERIFY	VER
20	OVERRIDE	OVE
21	ON	ON
22	OFF	OFF

	Command or Parameter	Abbreviation
23	NOW	NOW
24	QC	QC
25	ALL	ALL
26	DISPLAY	DIS
27	MODIFY	MOD
28	TOTALS	TOT
29	INQUIRY	INQ
30	DS	DS

F.5 COMMANDS AND ABBREVIATIONS USED BY FORMAINT

	Command or Parameter	Abbreviation
1	INSERT	INS
2	LIST	LIS
3	AMEND	AME
4	DELETE	DEL
5	DISPLAY	DIS
6	SAVE	SAV
7	GET	GET
8	REMOVE	REM
9	PD	PD
10	PRINT	PRI
11	HELLO	HEL
12	BYE	BYE
13	TERMINATE	TER
14	DUMP	DUM
15	LOAD	LOA
16	OPTIONS	OPT
17	SS	SS
18	GROUP	GRO
19	FORMAT	FOR
20	FIELD	FIE

APPENDIX G

The following is a list of internal and external file names by program:

	INTERNAL NAME	EXTERNAL NAME	TYPE
ODESY/FILEMAINT			
	SYSTEM	ODESYDATA/SYSTEM	Disk
	TANK	ODESYDATA/TANK	Disk
	FORMATS	ODESYDATA/FORMATS	Disk
	LINE	LINE	Printer
	COPY	COPY	Disk
	FORCOPY	FORCOPY	Disk
	WORK	ODESYWORK / #	Disk
	INPUT	INPUT	Disk
ODESY/FORMAINT			
	LOG	LOG	Printer
	LINE	LINE	Printer
	FORMATS	ODESYDATA/FORMATS	Disk
	SYSTEM	ODESYDATA/SYSTEM	Disk
	TANK	ODESYDATA/TANK	Disk
	DUMP	DUMP	Disk
	RMTE	STATIONS	Remote
ODESY/EDIT			
	SYSTEM	ODESYDATA/SYSTEM	Disk
	TANK	ODESYDATA/TANK	Disk
	FORMATS	ODESYDATA/FORMATS	Disk
	STATIONS	STATIONS	Remote
	LINE	LINE	Printer
	OPSTATS	ODESYDATA/OPSTATS	Disk

	INTERNAL NAME	EXTERNAL NAME	TYPE
ODESY/EXTRACT			
	LINE	LINE	Printer
	LOG	LOG	Printer
	FORMATS	ODESYDATA/FORMATS	Disk
	SYSTEM	ODESYDATA/SYSTEM	Disk
	TANK	ODESYDATA/TANK	Disk
	OPUT	OUTPUT	Disk
	COPY	OUTPUT	Disk
	OPSTATS	ODESYDATA/OPSTATS	Disk
	STATIONS	STATIONS	Remote
ODESY/STATS			
	LINE	LINE	Printer
	FORMATS	ODESYDATA/FORMATS	Disk
	SYSTEM	ODESYDATA/SYSTEM	Disk
ODESY/PRINT			
	LINE	LINE	Printer
	TANK	ODESYDATA/TANK	Disk
	SYSTEM	ODESYDATA/SYSTEM	Disk

Note: The name of the network controller which is ZIP-executed by FORMAINT and the EDIT is taken from the System file.

	INTERNAL NAME	EXTERNAL NAME	TYPE
ODESY/DUMP			
	LINE	LINE	Printer
	LOG	LOG	Printer
	FORMATS	ODESYDATA/FORMATS	Disk
	SYSTEM	ODESYDATA/SYSTEM	Disk
	TANK	ODESYDATA/TANK	Disk
ODESY/CONVERT			
	LINE	LINE	Printer
	DUMP	DUMP	Disk
	SYSTEM	ODESYDATA/SYSTEM	Disk

APPENDIX H - GLOSSARY OF TERMS

APPLICATION FILE

A file which is suitable for input to an application. An extracted file.

APPLICATION

A program or group of programs which is the ultimate user of the data produced by a data entry system.

AUDIT ENTRY CHECKS

The checks that are applied to data entered from a station to ensure its correctness, e.g. check digit verification, mandatory, numeric or alphabetic data and cross-footing.

BATCH

A group of related records which constitute a logical grouping for processing.

BATCH FILE

See APPLICATION FILE

BATCH HEADER

The first record in an application file which contains identifying information and control totals.

BATCH TOTALS

Control totals entered by the operator and generated by the computer which must match. Used to assure completeness of a batch.

CONSOLE

The device which allows the central machine operator to communicate with the M.C.P.

CONSTANTS

In relation to a field, constants are values which are to be unconditionally supplied by the EDIT rather than entered by the operator. Constants may also be used for range checks and value checks.

CONTROLLING REMOTE FILE

The Controlling Remote File is the last file with the Headers Option set to which the station in question has been assigned. There is not necessarily a C.R.F. If the Primary Remote File was opened with "Headers" then it is the C.R.F. If not, and there is a Secondary Remote File, then it becomes the C.R.F.

C.R.F.

See Controlling Remote File

CURSOR

The pointer character which indicates on the screen of a TD the position of the next character to be entered.

EXTRACTION

The process by which the batches held on the Tank file in the form convenient for the Data Entry System are converted to the form expected by the application system. It involves remapping fields, justifying fields and providing zero and blank fill.

FIELD

The smallest unit of data processed by the EDIT. It is a string of 1 to N characters comprising one logical unit such as an account number or order number.

FOLLOWING FORMAT

A format which will automatically follow the present one upon the successful entry of data.

FORMAT

A collection of field specifications which completely defines the input, display, journal and output of data in the system.

FORMS DELIMITERS

The characters which are placed around a portion of the screen of a TD700 or TD800 to define an input (or unprotected) area while in forms mode. Usually the square bracket [ ] or the brace { } or RS and US ► ◄.

FORMS MODE

A mode of operation for TD700s and TD800s where certain areas of the screen are "protected" and data may only be entered in those areas defined by forms delimiters.

GROUP

A collection of formats providing input to a single application, i.e. payroll or accounts receivable.

HEADER OPTION

If a program is to perform the MCS-type functions of message switching and network control it must open its Remote File with the Header Option specified. This gives it access to 50 bytes of additional information about each message received by the program and allows it to send network control commands to the network controller.

HOME

The upper,left — most character position on the screen of a TD.

INPUT SPECIFICATIONS

The part of a format which specifies the audit entry checks to be applied to incoming data.

JOURNAL SPECIFICATION

The part of a format which specifies the printing of the contents of a record at extraction time.

LINE

The wire that connects one or more terminals to the central computer.

LINE ADAPTER

An electronic interface between the wire that comes from the terminals (the line) and the Single Line Control (SLC) or Multi Line Control (MLC).

LINE CONTROL DISCIPLINE

The Line Control Discipline operates at two levels. At the lower level the Request Set, comprised of a Receive Request and a Transmit Request, build and transmit the poll and select strings and the actual messages to a station, and they also decode and detect errors in incoming messages. The Control, operating at a higher level, decides which stations are serviced and in which order. It also has the ability to change Request Sets for a station at the command of an MCS, for diagnostic reasons or to handle a change in the network.

LINE-TERMINAL-ACTIVITY INDICATOR

This is a light, usually located at the top of the keyboard on a TD, which flickers to indicate that the terminal is being polled by the central machine.

LTAI

See Line-Terminal-Activity Indicator

MCS

Message Control System:- A program which controls the flow of messages in the system and which can control the logical attributes of the system.

MESSAGE

The smallest unit processed by the data communications subsystem. It is a string of characters preceded by an identifying header and terminated by an ETX character.

MLC

See Multi Line Control.

MODE CONTROL

The programmatic control of the TD700 or TD800 which allows the TD to be left in receive mode (ready to receive another message) after the receipt of a transmission. The DC1 character for the TDs.

MULTI LINE CONTROL

An electronic interface between the computer and the terminals. A Multi Line Control can handle more than one line, but each line must have a Line Adapter.

NDL

See NETWORK DEFINITION LANGUAGE.

NETWORK

A system which represents a series of points and their inter-connections. In a data communications network the points are the stations and the central machine and the connections are the wires between them.

NETWORK CONTROLLER

A program which performs the polling, selecting, retries, file opens and file closes related to stations and remote files. It is written in NDL.

NETWORK DEFINITION LANGUAGE

The compiler which produces a network controller and a NIF when given the parameters defining the network and the control procedures to be used.

NETWORK INFORMATION FILE

A table containing a definition of the current data comm. network. It is generated during the creation of the network controller by the NDL compiler.

NIF

See NETWORK INFORMATION FILE

OUTPUT SPECIFICATIONS

The part of a format which specifies the reformatting and remapping to be done on each record and field during extraction.

POLL

The process by which the central machine communicates with a station. It is in effect an enquiry to the station asking whether it has a message to send.

PRIMARY REMOTE FILE

In relation to any particular station the Primary Remote File is the last file to which the station has been attached. It is the file which receives, by default, all messages from the station. Messages beginning with the signal character go to the Secondary Remote File (q.v.) if one exists. The Primary Remote File may or may not have been opened with the option "HEADERS".

QUEUE

An ordered list of items to be processed. It is held in memory or on disk. A remote file is two queues of messages, one input and the other output, awaiting processing by the program or the network controller.

RECORD

Either an entry of header information and input data in the Tank file, or an entry of extracted data in the Application file.

RECOVERY

The process that a program goes through when, upon staring up, it detects that the termination of its last run was abnormal.

REMOTE FILE

The device through which a program communicates with the stations in the network. A program writes a message to a station in its remote file and the network controller transmits it to the proper terminal, or a terminal transmits a message to the network controller which places it in the proper remote file for that station. A single station may receive messages from more than one program, i.e. be defined in more than one remote file.

SCREEN IMAGE

A string of characters held in the output buffer of the Edit which will produce the desired display at a station. It is built from the input layout included in the format.

SCREEN SPECIFICATION

The part of a format which specifies the field length, any identifying label which is to appear before the field and cursor positioning during display.

SECONDARY REMOTE FILE

In relation to any particular station the Secondary Remote File is the next to last file to which a station has been assigned. When an application program opens a remote file a message is sent to all MCS-type programs that currently have control of any of the stations in that remote file. If an MCS-type program elects to detach those stations from itself and attach them to the new file the remote file of the MCS-type program becomes the Secondary

Remote File. That Secondary Remote File then receives all messages which begin with the Signal Character from those stations.

SELECT

The process through which the central machine sends a message to a station. It is, first, an enquiry to the station asking if it is ready to receive a message and second the actual transmission of the message if the station is ready to receive.

SIGNAL CHARACTER

The character which identifies that a particular message is for the Secondary Remote File. It is defined at file open or attach time by the C.R.F. and must appear as the first character in a message from a station.

SINGLE LINE CONTROL

An electronic interface between the computer and the terminals. A Single Line Control can only handle one line with its Line Adapter.

SLC

See Single Line Control.

SPO

See CONSOLE.

STATION

A point where data may enter or leave a network. Its physical characteristics are defined by its associated terminal type.

TANK FILE

A work file used by the Edit to hold all batches until they are released.

TERMINAL

A hardware device type which defines the physical characteristics of a station, i.e. TD701, TD802.

TIME OUT

A condition wherein the network controller does not receive a reply to an enquiry (a POLL or SELECT) sent to a station within a specified length of time. The station is regarded as "not ready" or it is retried, depending on the line control discipline used.

INDEX

A

/ACCEPT command 5-19

/AMEND command 4-4

B

Batch correction after release 5-15

Batch deletion 6-5

Batch status print program 8-1

Batch status print program block diagram 8-2

Batch total option 4-27

Blank fill option 4-31

Blocking factor option 4-17

/BREAK command 5-41

/BUG command 5-42

/BYE command

1. format maintenance 4-7
2. data entry 5.20

C

CDV schemes

insertion 3-13

explanation and examples Appendix E

CDV option 4-26

/CLOSE command 5-21

Command inclusion 2-7

Command modification 3-12

Command restriction 2-7

Configuration parameters 3-5, C1

Constants :

inserted 4-37

range check 4-37

value check 4-38

/CONTROL command 5-43

Crossfooting option 4-26

D

Data Control 1-7

Data entry 5-5

Data entry - worked example 5-51-5-57

Data verification 5-9

/DELETE command

1. format maintenance 4-4
2. data entry 5-22

/DISPLAY command 4-7

Display command (/) 5-23

/DS command 5-44

/DUMP command

1. format maintenance 4-7

2. control 5 45

Duplicate anything option 2-7

Duplicate In option 4-25

Duplicate Out option 4-30

E

Edit program (EDIT) 1-5, section 5

EDIT block diagram 5-4

EDIT commands

1. data entry 5-19 - 5-40

2. control 5-41 - 5-52

Emergency procedures 1-10

End line option 4-25

Enter (//) command 5-25

Entry of format data 4-14

Error text modification 3-11, C-11

Error messages Appendix F

Extraction program (EXTRACT) 1-5, section 6

Extraction and release 6-3

EXTRACT block diagram 6-2

F

Field details 4-23

Field engineer settings for TD820, TD830 and TD730 terminals 2-11

Field length 4-24

Field name 4-23

Field sequence number 4-23

Field type 4-24

File maintenance program (FILEMAINT) 1-4, section 3

FILEMAINT block diagram 3-3

File name 4-16

Format header details 4-20

Format maintenance program (FORMAINT) 1-4, section 4

FORMAINT block diagram 4-3

Format maintenance commands

1. explanation 4-4 - 4-15
2. Summary Appendix B

Format name 4-20

Format type option 4-22

G

Generation of the network controller 2-9

/GET command 4-9

Group header details 4-16

Group name 4-16

H

/HELLO command

1. format maintenance 4-9
2. data entry 5-26

I

Implementation plan 2-1

Initialization of the files 3-4

Input audit options 1-2, section 4

Input data size 2-4

Input forms Appendix C

Inquiry (//) command 5-27

/INSERT command

1. format maintenance 4-4
2. data entry 5-28

Installation procedures section 2

J

Journal option 1-3, section 4

Justify in option 4-27

Justify out option 4-32

L

Label type option 4-18

/LIST command 4-4

/LOAD command 4-7

M

Mandatory field option 4-27

Medium option 4-18

Modify (//) command 5-29

N

Next format option 4-20

Number of areas option 4-17

Number of characters of constant data (EDIT) 2-5

Number of characters of scratchpad constant data (FORMAINT) 2-5

Number of fields (EDIT) 2-5

Number of formats (EDIT) 2-5

Number of fields (FORMAINT) 2-5

Number of fields (EXTRACT) 2-6

Number of formats (EXTRACT) 2-6

O

/OPEN command 5-30

Opening a batch 5-7

Operator parameters 3-10, C-9

/OPTIONS command 4-9

Output options 1-3, section 4

Output position option 4-28

Output reports Appendix D

P

Packed option 4-31

Print after option 4-34

Print column option 4-34

/PRINT command 4-11

/PD (print directory) command 4-10

Print facilities 3-14

Print name option 4-35

R

/READY command 5-46

Reconfiguration of the ODESY system 2-18

Record size option 4-16

Record size for the Tank file 2-3

Recording mode option 4-18

Records per area option 4-17

Recovery 1-8, 6-7

/RELEASE command 5-33

Releasing a batch 5-12

/REMOVE command 4-11

Repeat (//) command 4-12

Report option 4-17

Report input record number 4-19

Report output record number 4-19

Running ODESY initially 2-14

S

/SAVE command 4-12

Screen display options 1-2, section 4

Security 1-9

Security level 4-21

Selecting system features 2-3

Sign option 4-32

Sign convention option 4-33

SS (send to station) command

1. format maintenance 4-12
2. control 5-47

Standard system configuration 2-12

Station buffer size (FORMAINT) 2-6

Station parameters 3-9, C-7

Statistics analysis program (STATS) 1-5, section 7

STATS block diagram 7-3

STATUS command 5-36

System software 1-4

System hardware 1-6

T

/TERMINATE command

1. format maintenance 4-7
2. control 5-48

Typical network controller listing 2-10

V

Verify (//) command 5-39

Verify option 4-21

W

Worked example 1 (format maintenance) 4-39

Worked example 2 (format maintenance) 4-44

Write after option 4-28

/WRU (WHO ARE YOU) command 5-50